



PicoScope® 6

PC Oscilloscope Software

User's Guide



S J ELECTRONICS
POWER • TEST & MEASUREMENT

☎ 0800 583 4455
✉ sales@sjelectronics.co.uk

VISIT ⇌
WEBSITE



Table of Contents

| | |
|--|-----------|
| 1 Welcome | 1 |
| 2 PicoScope 6 overview | 2 |
| 3 Introduction | 3 |
| 1 Legal statement | 3 |
| 2 Upgrades | 3 |
| 3 Trade marks | 4 |
| 4 System requirements | 4 |
| 4 Using PicoScope for the first time | 5 |
| 5 PicoScope and oscilloscope primer | 6 |
| 1 Oscilloscope basics | 6 |
| 2 PC Oscilloscope basics | 7 |
| 3 PicoScope basics | 7 |
| 1 Capture modes | 8 |
| 2 How do capture modes work with views? | 9 |
| 4 PicoScope window | 10 |
| 5 Scope view | 11 |
| 6 Overrange indicator | 12 |
| 7 MSO view | 13 |
| 1 Digital view | 14 |
| 2 Digital context menu | 15 |
| 8 XY view | 16 |
| 9 Trigger marker | 17 |
| 10 Post-trigger arrow | 17 |
| 11 Spectrum view | 18 |
| 12 Persistence mode | 19 |
| 13 Measurements table | 20 |
| 14 Pointer tool tip | 21 |
| 15 Signal rulers | 22 |
| 16 Time rulers | 23 |
| 17 Phase (rotation) rulers | 24 |
| 18 Ruler settings | 26 |
| 19 Ruler legend | 27 |
| 20 Frequency legend | 27 |
| 21 Properties sheet | 28 |
| 22 Custom probes | 29 |
| 23 Math channels | 30 |
| 24 Reference waveforms | 31 |
| 25 Serial decoding | 32 |
| 26 Mask limit testing | 33 |
| 27 Alarms | 34 |
| 28 Buffer Overview | 35 |

| | |
|---|-----|
| 6 Menus | 36 |
| 1 File menu | 37 |
| 1 Save As dialog | 38 |
| 2 Startup Settings menu | 45 |
| 3 Waveform Library Browser | 46 |
| 2 Edit menu | 47 |
| 1 Notes | 48 |
| 2 Channel Labels (PicoScope Automotive only) | 48 |
| 3 Details dialog (PicoScope Automotive only) | 49 |
| 3 Views menu | 50 |
| 1 Custom grid layout dialog | 51 |
| 4 Measurements menu | 52 |
| 1 Add / Edit Measurement dialog | 53 |
| 2 Advanced measurement settings | 54 |
| 5 Tools menu | 55 |
| 1 Custom Probes dialog | 56 |
| 2 Math Channels dialog | 70 |
| 3 Reference Waveforms dialog | 81 |
| 4 Serial Decoding dialog | 83 |
| 5 Alarms dialog | 89 |
| 6 Masks menu | 91 |
| 7 Macro Recorder | 94 |
| 8 Preferences dialog | 95 |
| 6 Help menu | 110 |
| 7 Automotive menu (PicoScope Automotive only) | 111 |
| 8 Connect Device dialog | 112 |
| 9 Demo mode | 113 |
| 10 Converting files in Windows Explorer | 114 |
| 7 Toolbars and buttons | 116 |
| 1 Advanced Options toolbar | 116 |
| 2 Channels toolbar | 117 |
| 1 Channel Options menu | 118 |
| 2 ConnectDetect | 124 |
| 3 Digital Inputs button | 125 |
| 3 PicoLog 1000 Series Channels toolbar | 128 |
| 1 PicoLog 1000 Series Digital Outputs control | 129 |
| 4 USB DrDAQ Channels toolbar | 130 |
| 1 USB DrDAQ RGB LED control | 131 |
| 2 USB DrDAQ Digital Outputs control | 132 |
| 5 Capture Setup toolbar | 133 |
| 1 Spectrum Options dialog | 135 |
| 2 Persistence Options dialog | 137 |
| 6 Buffer Navigation toolbar | 139 |

| | |
|---|-----|
| 7 Measurements toolbar | 140 |
| 8 Signal Generator button | 141 |
| 1 Signal Generator dialog (PicoScope devices) | 141 |
| 2 Signal Generator dialog (USB DrDAQ) | 144 |
| 3 Arbitrary waveform files | 145 |
| 4 Arbitrary Waveform Generator window | 146 |
| 5 Demo Signals menu | 150 |
| 6 Demo Signals dialog | 151 |
| 9 Start / Stop toolbar | 152 |
| 10 Triggering toolbar | 153 |
| 1 Advanced Triggering dialog | 155 |
| 2 Advanced trigger types | 156 |
| 11 Zooming and Scrolling toolbar | 162 |
| 1 Zoom Overview window | 163 |
| 8 How to... | 164 |
| 1 How to change to a different device | 164 |
| 2 How to use rulers to measure a signal | 165 |
| 3 How to measure a time difference | 166 |
| 4 How to move a view | 167 |
| 5 How to scale and offset a signal | 168 |
| 6 How to set up the spectrum view | 170 |
| 7 How to find a glitch using persistence mode | 171 |
| 8 How to set up a Mask Limit Test | 175 |
| 9 How to save on trigger | 178 |
| 10 How to create a link file | 182 |
| 9 Reference | 184 |
| 1 Measurement types | 184 |
| 1 Scope measurements | 184 |
| 2 Spectrum measurements | 185 |
| 2 Signal generator waveform types | 187 |
| 3 Spectrum window functions | 188 |
| 4 Trigger timing (part 1) | 189 |
| 5 Trigger timing (part 2) | 190 |
| 6 Serial protocols | 191 |
| 1 1-Wire protocol | 192 |
| 2 ARINC 429 protocol | 192 |
| 3 CAN protocol | 193 |
| 4 DCC protocol | 193 |
| 5 DMX512 protocol | 194 |
| 6 Ethernet 10BASE-T protocol | 194 |
| 7 Fast Ethernet 100BASE-TX protocol | 195 |
| 8 FlexRay protocol | 196 |
| 9 I ² C protocol | 197 |

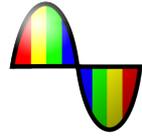
| | |
|------------------------------------|-----|
| 10 I ² S protocol | 197 |
| 11 LIN protocol | 198 |
| 12 PS/2 protocol | 198 |
| 13 SENT Fast protocol | 199 |
| 14 SENT Slow protocol | 200 |
| 15 SPI protocol | 200 |
| 16 UART protocol | 201 |
| 17 USB protocol | 201 |
| 7 Device feature table | 202 |
| 8 Command-line syntax | 204 |
| 9 Flexible power | 207 |
| 10 Glossary | 208 |
| Index | 211 |

1 Welcome

Welcome to **PicoScope 6**, the PC Oscilloscope software from Pico Technology.

With a scope device from Pico Technology, [PicoScope](#) turns your PC into a powerful [PC Oscilloscope](#) with all the features and performance of a benchtop [oscilloscope](#) at a fraction of the cost.

- [Key features of this version](#)
- [Using PicoScope for the first time](#)



2 PicoScope 6 overview

PicoScope 6 is Pico Technology's software for PC Oscilloscopes.

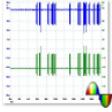
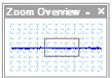
● High performance

- Fast capture rates, making it easier to see fast-moving signals
- Fast data processing
- Support for the latest PicoScope USB oscilloscopes
- Frequent free-of-charge updates

● Advanced usability and display

- Clear graphics and text
- Tool tips and help messages to explain all features
- Easy point-and-click tools for panning and zooming
- Quick save, print, and file sharing capability

● Key features

| | | | |
|---|---|---|--|
|  | The latest Windows .NET technology enabling us to deliver updates more quickly |  | Multiple views of the same data, with individual zoom and pan settings for each view |
|  | Custom probes manager to make it easy for you to use your own probes and sensors with PicoScope |  | Advanced triggering conditions including pulse, window and logic |
|  | Properties sheet displaying all settings at a glance |  | Spectrum mode with a fully optimized spectrum analyzer |
| 1 kHz | Per-channel lowpass filtering |  | Math channels for creating mathematical functions of input channels |
|  | Reference waveforms for storing copies of input channels |  | Arbitrary waveform designer for oscilloscopes with a built-in arbitrary waveform generator |
|  | Rapid trigger mode to capture a sequence of waveforms with the minimum possible dead time |  | Windows Explorer integration to show files as pictures and convert to other formats |
|  | Command-line options for converting files |  | Zoom overview for rapidly adjusting the zoom to show any part of the waveform |
|  | Serial decoding for UART, I ² C and other formats, in real time |  | Mask limit testing to show when a signal goes out of bounds |
|  | Buffer overview for searching the waveform buffer |  | Alarms to alert you when a specified event occurs |

3 Introduction

PicoScope 6 is a comprehensive software application for Pico Technology oscilloscopes. Used with a PicoScope hardware device, it creates an oscilloscope and spectrum analyzer on your PC.

PicoScope 6 supports the devices listed in the [Device feature table](#). It runs on Windows computers (see [System requirements](#) for further recommendations).

How to use PicoScope 6

- Getting started: see [using PicoScope for the first time](#), and PicoScope's [Features](#).
- For further information: see descriptions of [Menus](#) and [Toolbars](#), and the [Reference](#) section.
- For step-by-step tutorials, see the [How to](#) section.

3.1 Legal statement

Grant of license. The material contained in this release is licensed, not sold. Pico Technology Limited ('Pico') 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 and agree to abide by these conditions.

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

Copyright. Pico claims the copyright of, and retains the rights to, all material (software, documents etc.) contained in this release.

Liability. Pico and its agents shall not be liable for any loss or damage, 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, so Pico cannot guarantee that its equipment or software is suitable for a given application. It is therefore the user's responsibility to ensure that the product is suitable for the user's application.

Mission-critical applications. Because the software runs on a computer that may be running other software products, and may be subject to interference from these other products, this license specifically excludes usage in 'mission-critical' applications, for example life-support systems.

Viruses. This software was continuously monitored for viruses during production. However, the user is responsible for virus checking the software once it is installed.

Support. No software is ever error-free, but if you are dissatisfied with the performance of this software, please contact our technical support staff.

3.2 Upgrades

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

3.3 Trade marks

Windows is a registered trade mark of Microsoft Corporation. *Pico Technology*, *PicoScope*, *PicoLog*, *DrDAQ*, and *ConnectDetect* are internationally registered trade marks.

3.4 System requirements

To ensure that PicoScope operates correctly, you must have a computer with at least the minimum system requirements to run your Windows operating system, which must be one of the versions listed in the following table. The performance of the oscilloscope will be better with a more powerful PC, and will benefit from a multi-core processor.

| Item | Specification |
|-------------------------|--|
| Operating system | Windows 7, Windows 8 (not Windows RT), Windows 10*. 32 bit and 64 bit versions. Beta software is also available for Linux and OS X operating systems. |
| Processor | As required by the operating system |
| Memory | |
| Free disk space | |
| Ports | USB 3.0 or USB 2.0 port(s) |

* PicoScope version 6.11 and SDKs are compatible with Windows XP SP3 and Vista SP2 in addition to the Windows versions listed above. For best performance we recommend Windows 7 or later.

4 Using PicoScope for the first time

We have designed PicoScope to be as easy as possible to use, even for newcomers to oscilloscopes. Once you have followed the introductory steps listed below, you will soon be on your way to becoming a PicoScope expert.

-  1. Install the software. Load the CD-ROM that is included with your scope device, then click the **Install Software** link and follow the on-screen instructions.
-  2. Plug in your scope device. Windows will recognise it and prepare your computer to work with it. Wait until Windows tells you that the device is ready to use.
-  3. Click the new PicoScope icon on your Windows desktop.
-  4. PicoScope will detect your scope device and prepare to display a waveform. The green [Start](#) button will be highlighted to show that PicoScope is ready.
-  5. Connect a signal to one of the scope device's input channels and see your first waveform! To learn more about using PicoScope, please read the [PicoScope Primer](#).

Problems?

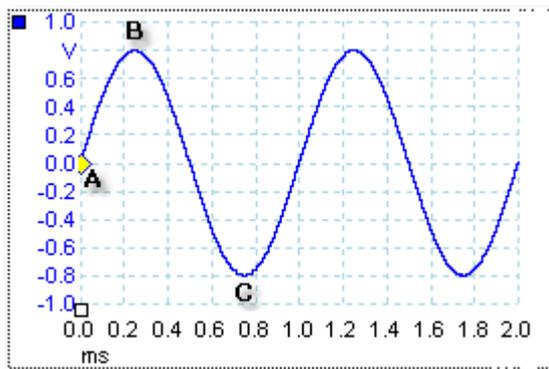
Help is at hand! Our technical support staff are always ready to answer your telephone call during office hours (09:00 to 17:00, UK time, Mon-Fri: see our Contact Details). At other times, you can leave a message on our [support forum](#) or [send us an email](#).

5 PicoScope and oscilloscope primer

This chapter explains the fundamental concepts that you will need to know before working with the PicoScope software. If you have used an oscilloscope before, then most of these ideas will be familiar to you. You can skip the [Oscilloscope basics](#) section and go straight to the [PicoScope-specific information](#). If you are new to oscilloscopes, please take a few minutes to read at least the [Oscilloscope basics](#) and [PicoScope basics](#) topics.

5.1 Oscilloscope basics

An **oscilloscope** is a measuring instrument that displays a graph of voltage against time. For example, the picture below shows a typical display on an oscilloscope screen when a varying voltage is connected to one of its input channels.



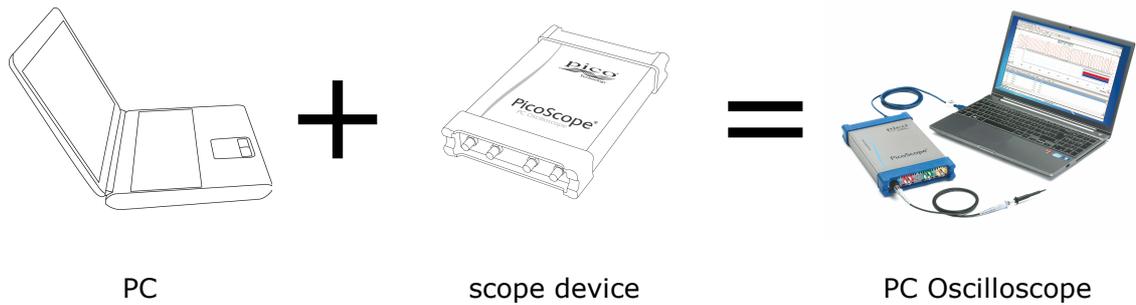
Oscilloscope displays are always read from left to right. The voltage-time characteristic (waveform) of the signal is drawn as a line called the **trace**. In this example, the trace is blue and begins at point **A**. If you look to the left of this point, you will see the number **0.0** on the voltage [axis](#), which tells you that the voltage is 0.0 V (volts). If you look below point **A**, you will see another number **0.0**, this time on the time axis, which tells you that the time is 0.0 ms (milliseconds) at this point.

At point **B**, 0.25 milliseconds later, the voltage has risen to a positive peak of 0.8 volts. At point **C**, 0.75 milliseconds after the start, the voltage has dropped to a negative peak of -0.8 volts. After 1 millisecond, the voltage has risen back to 0.0 volts and a new cycle is about to begin. This type of signal is called a sine wave, and is one of a limitless range of signal types that you will encounter.

Most oscilloscopes allow you to adjust the vertical and horizontal scales of the display. The vertical scale is called the **voltage range** (in this example at least, although scales in other units, such as milliamperes, are possible). The horizontal scale is called the **timebase** and is measured in units of time - in this example, thousandths of a second.

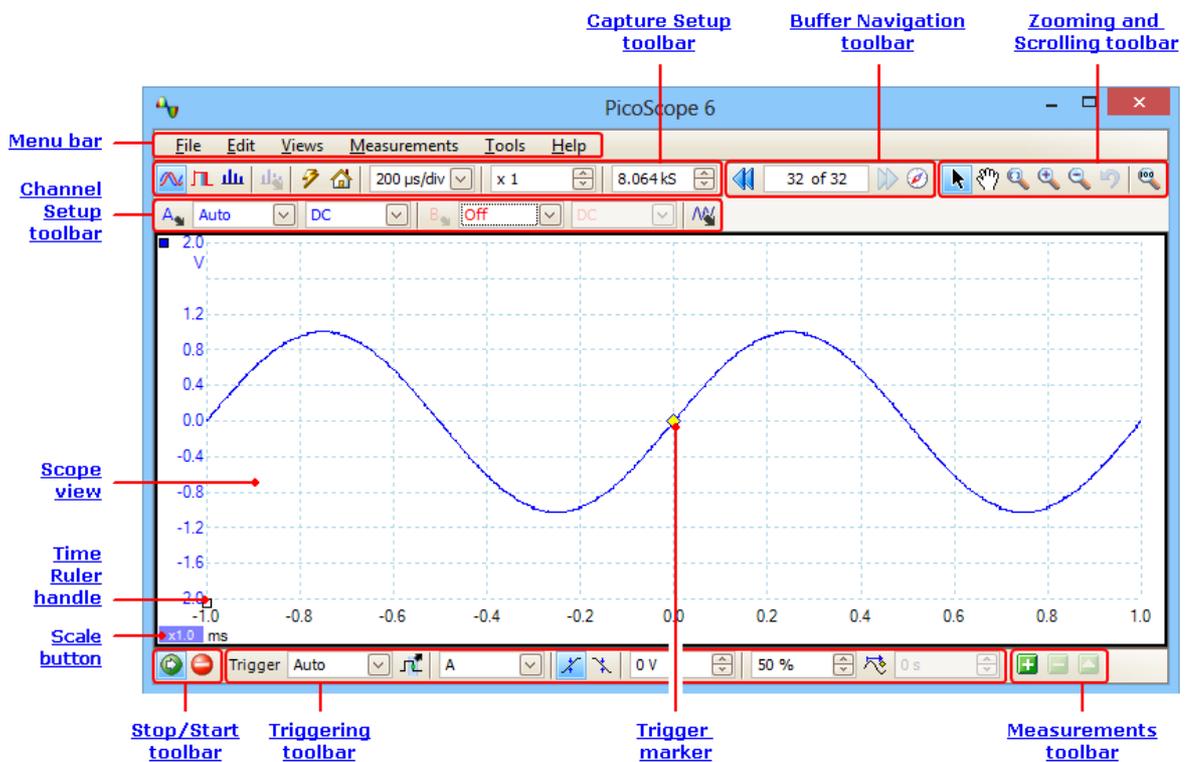
5.2 PC Oscilloscope basics

A **PC Oscilloscope** is a measuring instrument that consists of a hardware scope device and an oscilloscope program running on a PC. Oscilloscopes were originally stand-alone instruments with no signal processing or measuring abilities, and with storage only available as an expensive extra. Later oscilloscopes began to use new digital technology to introduce more functions, but they remained highly specialised and expensive instruments. **PC Oscilloscopes** are the latest step in the evolution of oscilloscopes, combining the measuring power of Pico Technology's scope devices with the convenience of the PC that's already on your desk.



5.3 PicoScope basics

PicoScope can produce a simple display such as the example in the [Oscilloscope basics](#) topic, but it also has many advanced features. The screen shot below shows the PicoScope window. Click on any of the underlined labels to learn more. See [PicoScope window](#) for an explanation of these important concepts.



Note: Other buttons may appear in the PicoScope main window depending on the capabilities of the oscilloscope that is connected, and on the settings applied to the PicoScope program.

5.3.1 Capture modes

PicoScope can operate in three capture modes: **scope mode**, **spectrum mode** and **persistence mode**. The mode is selected by buttons in the [Capture Setup Toolbar](#).

Capture mode buttons



- In **scope mode**, PicoScope displays a main [scope view](#), optimizes its settings for use as a PC Oscilloscope, and allows you to directly set the capture time. You can still display one or more secondary spectrum views.
- In **spectrum mode**, PicoScope displays a main [spectrum view](#), optimizes its settings for spectrum analysis, and allows you to directly set the frequency range in a similar way to a dedicated spectrum analyzer. You can still display one or more secondary scope views.
- In **persistence mode**, PicoScope displays a single, modified scope view in which old waveforms remain on the screen in faded colors while new waveforms are drawn in brighter colors. See also: [How to find a glitch using persistence mode](#) and the [Persistence Options dialog](#).

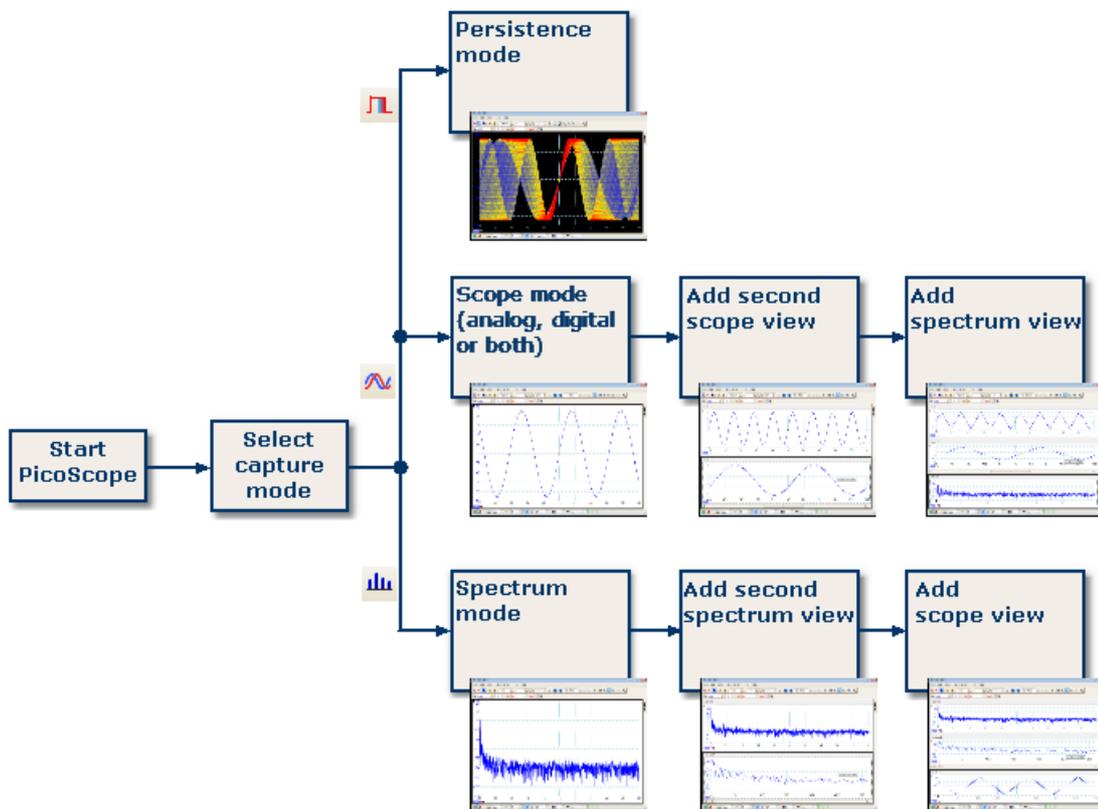
When you [save waveforms and settings](#), PicoScope only saves data for the mode that is currently in use. If you wish to save settings for both capture modes, then you need to switch to the other mode and save your settings again.

See also: [How do capture modes work with views?](#)

5.3.2 How do capture modes work with views?

The [capture mode](#) tells PicoScope whether you are mainly interested in viewing waveforms ([scope mode](#)) or frequency plots ([spectrum mode](#)). When you select a capture mode, PicoScope sets up the hardware appropriately and then shows you a **view** that matches the capture mode (a [scope view](#) if you selected scope mode or [persistence mode](#), or a [spectrum view](#) if you selected spectrum mode). The rest of this section does not apply in persistence mode, which allows only a single view.

Once PicoScope has shown you the first view, you can, if you wish, add more scope or spectrum views, regardless of the capture mode you are in. You can add and remove as many extra views as you wish, as long as one view remains that matches the capture mode.

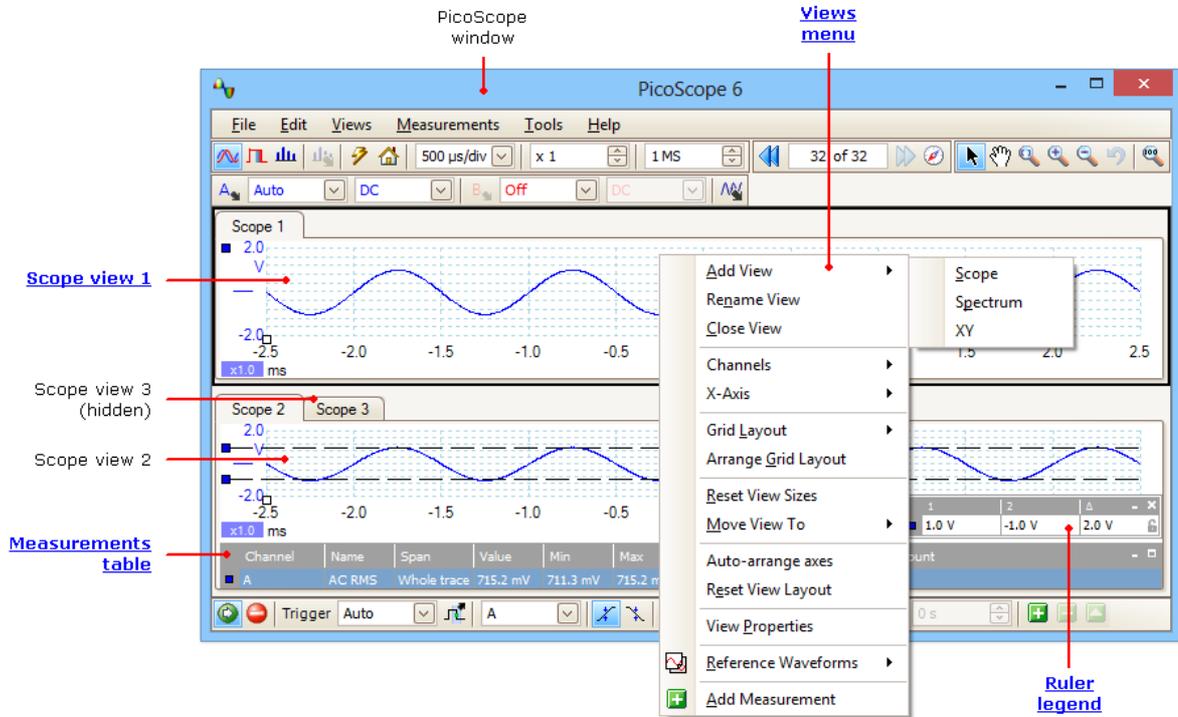


Examples showing how you might select the capture mode and open additional views in PicoScope. Top: persistence mode (one view only). Middle: scope mode. Bottom: spectrum mode.

When using a secondary view type (a spectrum view in scope mode, or a scope view in spectrum mode), you may see the data compressed horizontally rather than displayed neatly as in a primary view. You can usually overcome this by using the zoom tools.

5.4 PicoScope window

The **PicoScope window** shows a block of data captured from the [scope device](#). When you first open PicoScope it contains one [scope view](#), but you can add more views by clicking **Add view** in the [Views menu](#). The screen shot below shows all the main features of the PicoScope window. Click on the underlined labels for more information.



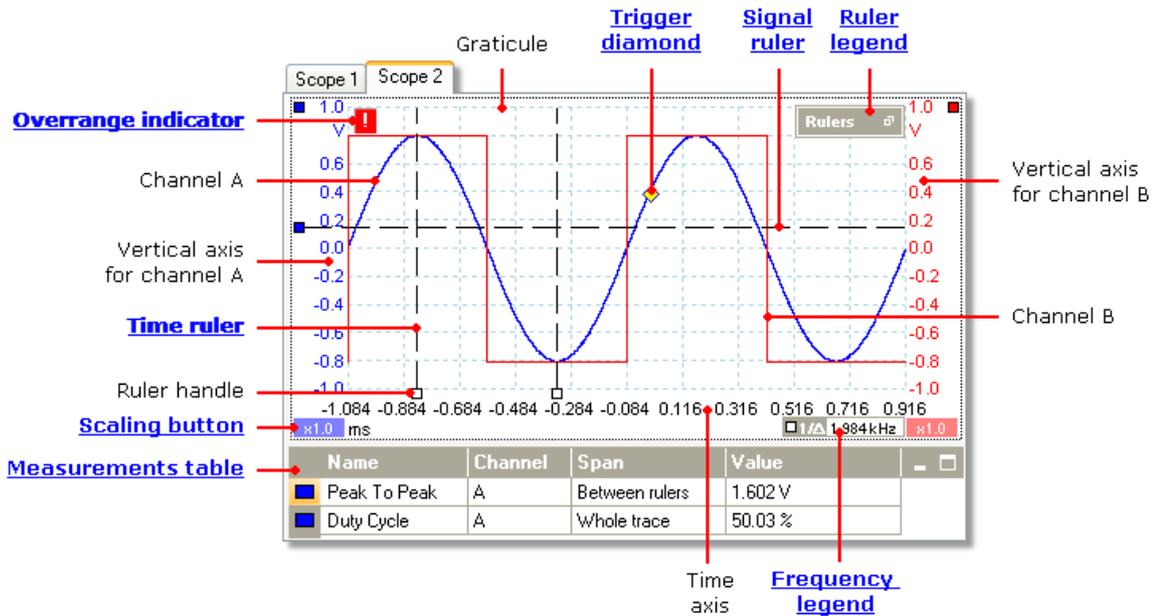
To arrange the views within the PicoScope window

If the PicoScope window contains more than one [view](#), PicoScope arranges them in a grid. This is arranged automatically, but you can customize it if you wish. Each rectangular space in the grid is called a [viewport](#). You can move a [view](#) to a different viewport by dragging its name tab ([show me](#)), but you cannot move it outside the PicoScope window. You can also put more than one view in a viewport, by dragging a view and dropping it on top of another.

For further options, right-click on a view to obtain the [View menu](#), or select **View** from the [Menu bar](#), then select one of the menu options to arrange the views.

5.5 Scope view

A **scope view** shows the data captured from the scope as a graph of signal amplitude against time. (See [Oscilloscope basics](#) for more on these concepts). PicoScope opens with a single view, but you can add more views by using the [views menu](#). Similar to the screen of a conventional oscilloscope, a scope view shows you one or more waveforms with a common horizontal time axis, with signal level shown on one or more vertical axes. Each view can have as many waveforms as the scope device has channels. Click on one of the labels below to learn more about a feature.



Scope views are available regardless of which mode - [scope mode](#) or [spectrum mode](#) - is active.

5.6 Overrange indicator

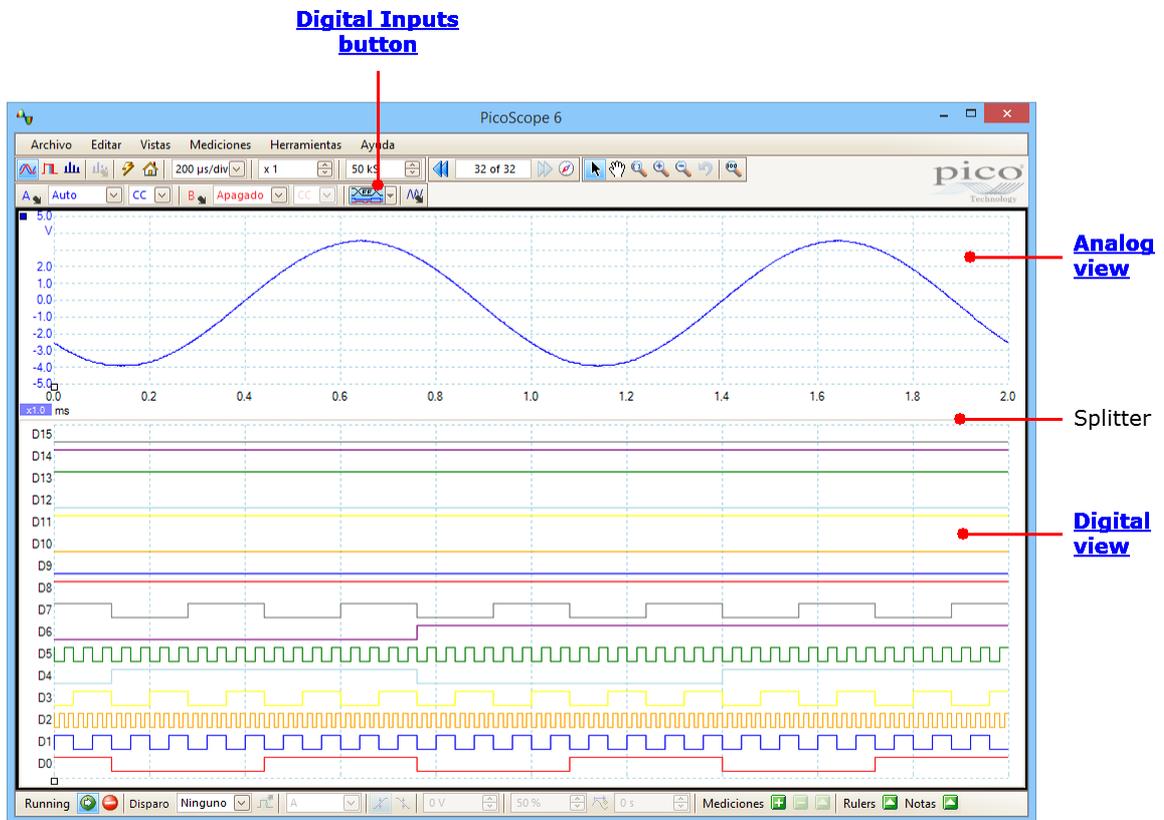
If an overrange (a signal outside the measuring range) is detected, the red warning icon (⚠) will appear in the upper corner of the PicoScope screen next to the relevant channel's vertical axis.

Scopes with floating inputs only: If the BNC shell to chassis voltage exceeds the measurement limit, the Channel LED will turn solid red, and the yellow warning icon (⚠) will appear in the upper corner of the PicoScope screen next to the relevant channel's vertical axis. Parts of the waveform will also be missing when the measurement limit is exceeded.

5.7 MSO view

Applicability: mixed-signal oscilloscopes ([MSOs](#)) only

The **MSO view** shows mixed analog and digital data on the same timebase.



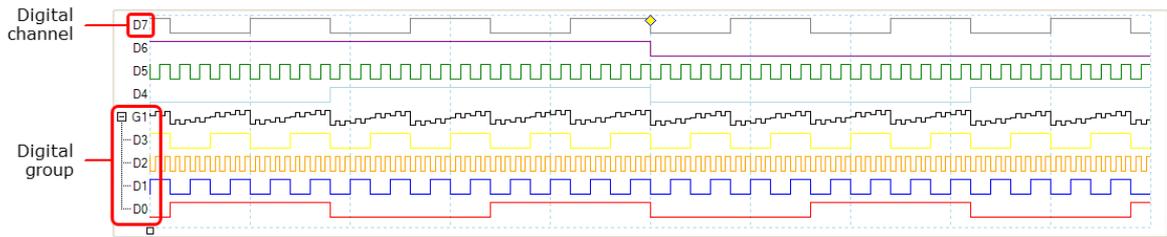
Digital Inputs button: Switches [digital view](#) on and off, and opens the [Digital Setup dialog](#).

Analog view: Shows the analog channels. The same as a standard [scope view](#).

Digital view: Shows the digital channels and groups. See [digital view](#).

Splitter: Drag up and down to move the partition between analog and digital sections.

5.7.1 Digital view

Location: [MSO view](#)

Note 1: You can right-click on the **digital view** to obtain the [Digital Context Menu](#).

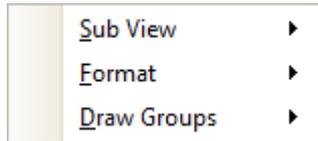
Note 2: If the **digital view** is not visible when required, check that (a) the [Digital Inputs button](#) is activated and (b) at least one digital channel is selected for display in the [Digital Setup dialog](#).

Digital channel: Displayed in the order in which they appear in the [Digital Setup dialog](#), where they can be renamed.

Digital group: Groups are created and named in the [Digital Setup dialog](#). You can expand and collapse them in the **digital view** using the and buttons.

5.7.2 Digital context menu

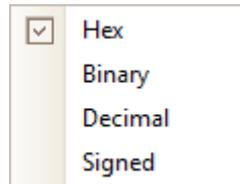
Location: right-click on the [digital view](#)

**Sub View:**

Analog: View or hide the [analog scope view](#).

Digital: View or hide the [digital scope view](#).

Also available from the [Views menu](#).

Format:

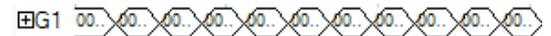
The numerical format in which group values are displayed in the [digital scope view](#).

Draw Groups:

By Values: Draw groups with transitions only where the value changes:



By Time: Draw groups with transitions spaced equally in time, once per sampling period. You will usually need to zoom in to see the individual transitions:

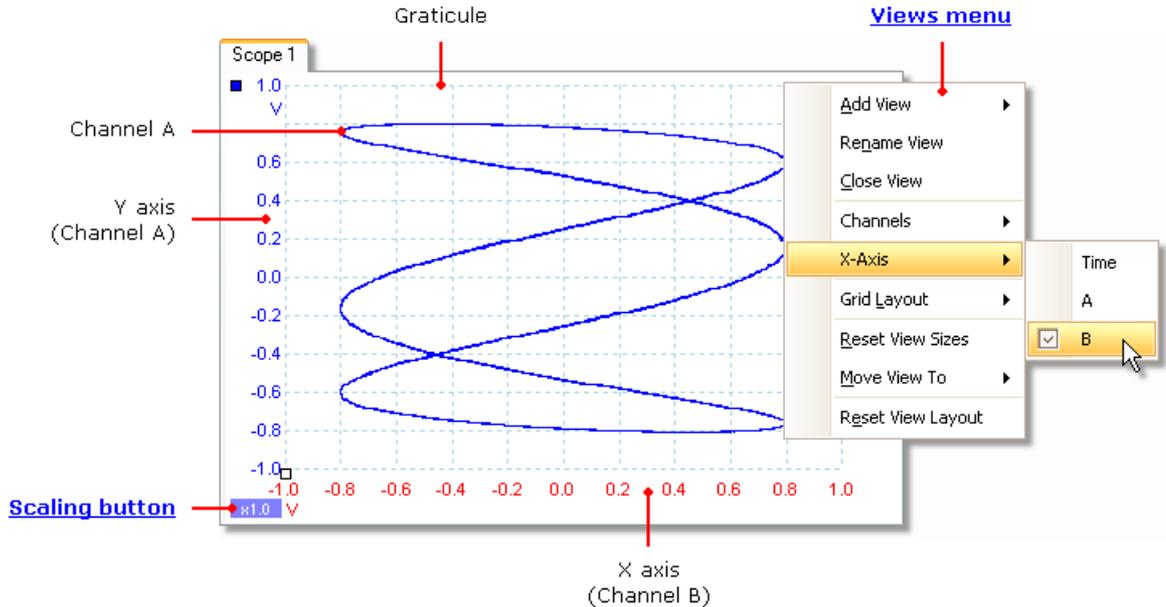


By Level: Draw groups as analog levels derived from the digital data:



5.8 XY view

An **XY view**, in its simplest form, shows a graph of one channel plotted against another. XY mode is useful for showing relationships between periodic signals (using Lissajous figures) and for plotting I-V (current-voltage) characteristics of electronic components.



In the example above, two different periodic signals have been fed into the two input channels. The smooth curvature of the trace tells us that the inputs are roughly or exactly sine waves. The three loops in the trace show that Channel B has about three times the frequency of Channel A. We can tell that the ratio is not exactly three because the trace is slowly rotating, although you cannot see that in this static picture. Since an XY view has no time axis, it tells us nothing about the absolute frequencies of the signals. To measure frequency, we need to open a [Scope view](#).

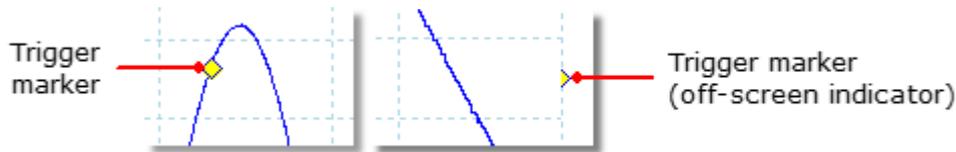
How to create an XY view

There are two ways to create an XY view.

- Use the **Add View > XY** command on the [Views menu](#). This adds a new XY view to the PicoScope window without altering the original [scope](#) or [spectrum](#) view or views. It automatically chooses the two most suitable channels to place on the X and Y axes. Optionally, you can change the X axis channel assignment using the **X-Axis** command (see below).
- Use the **X-Axis** command on the [Views menu](#). This converts the current scope view into an XY view. It maintains the existing Y axes and allows you to choose any available channel for the X axis. With this method, you can even assign a [math channel](#) or a [reference waveform](#) to the X axis.

5.9 Trigger marker

The **trigger marker** shows the level and timing of the trigger point.



The height of the marker on the vertical axis shows the level at which the trigger is set, and its position on the time axis shows the time at which it occurs.

You can move the trigger marker by dragging it with the mouse or, for more accurate control, by using the buttons on the [Triggering toolbar](#).

Other forms of trigger marker

If the scope view is zoomed and panned so that the trigger point is off the screen, the off-screen trigger marker (shown above) appears at the side of the graticule to indicate the trigger level.

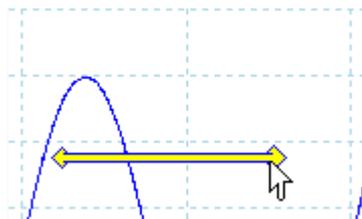
In post-trigger delay mode, the trigger marker is temporarily replaced by the [post-trigger arrow](#) while you adjust the post-trigger delay.

When some [advanced trigger types](#) are in use, the trigger marker changes to a window marker, which shows the upper and lower trigger thresholds.

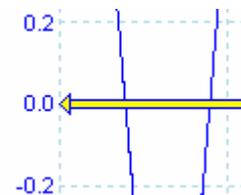
For more information, see the section on [Trigger timing](#).

5.10 Post-trigger arrow

The **post-trigger arrow** is a modified form of the [trigger marker](#) that appears temporarily on a [scope view](#) while you are setting up a post-trigger delay, or dragging the trigger marker after setting up a post-trigger delay. ([What is a post-trigger delay?](#))



The left-hand end of the arrow indicates the trigger point, and is aligned with zero on the time axis. If zero on the time axis is outside the [scope view](#), then the left-hand end of the post-trigger arrow appears like this:

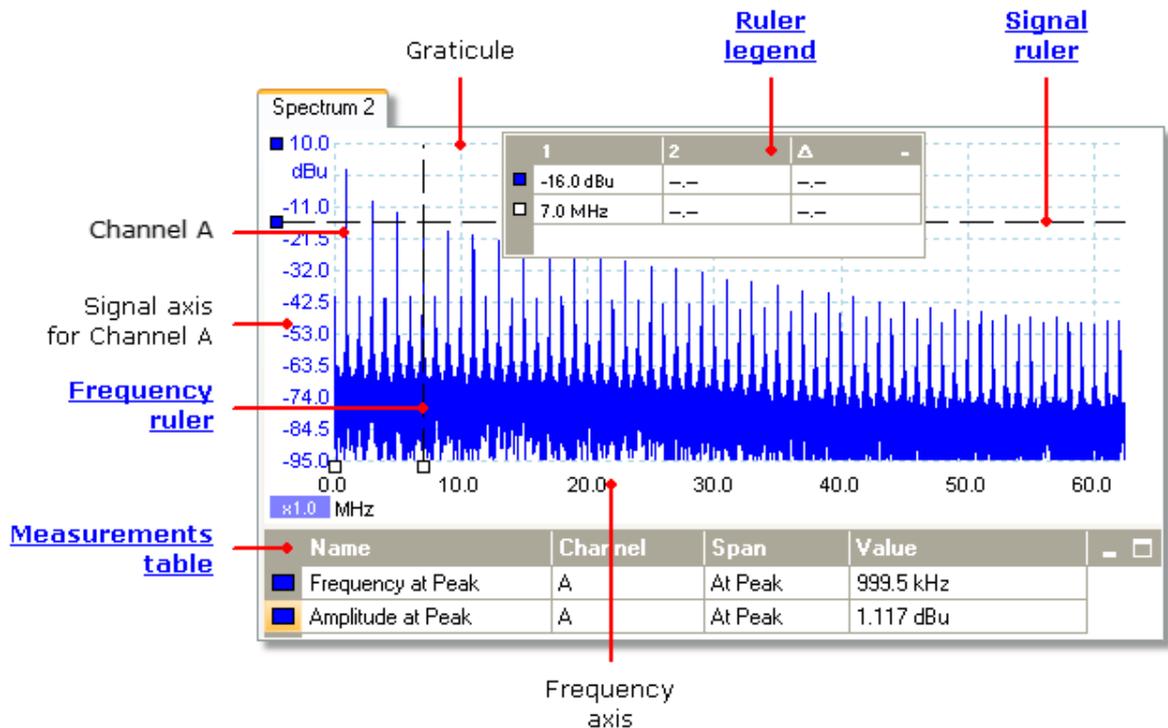


The right-hand end of the arrow (temporarily replacing the [trigger marker](#)) indicates the trigger reference point.

Use the buttons on the [Triggering toolbar](#) to set up a post-trigger delay.

5.11 Spectrum view

A **spectrum view** is one view of the data from a scope device. A spectrum is a diagram of signal level on a vertical axis plotted against frequency on the horizontal axis. PicoScope opens with a scope view, but you can add a spectrum view by using the [views menu](#). Similar to the screen of a conventional spectrum analyzer, a spectrum view shows you one or more spectra with a common frequency axis. Each view can have as many spectra as the scope device has channels. Click on one of the labels below to learn more about a feature.



Unlike in the scope view, in the spectrum view the data is not clipped at the limits of the range displayed on the vertical axis, so you can apply axis scaling or offset to see more data. Vertical axis labels are not provided for data outside what is considered to be the 'useful' range, but rulers will still work outside this range.

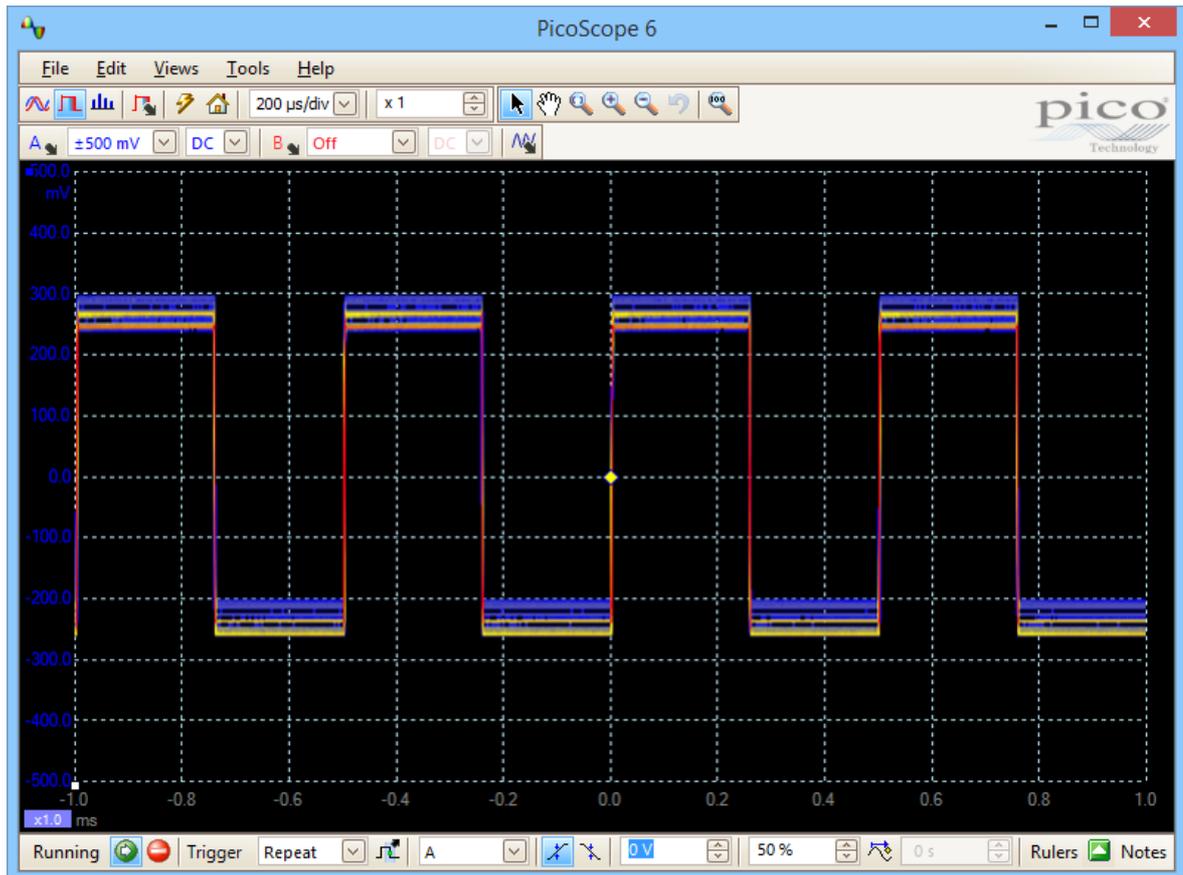
Spectrum views are available regardless of which mode - [Scope Mode](#) or [Spectrum Mode](#) - is active.

For more information, see: [How to set up the spectrum view](#) and [Spectrum Options dialog](#).

5.12 Persistence mode

Persistence mode superimposes multiple waveforms on the same view, with more frequent data or newer waveforms drawn in brighter colors than older ones. This is useful for spotting glitches, when you need to see a rare fault event hidden in a series of repeated normal events.

Enable persistence mode by clicking the **Persistence Mode button**  on the [Capture Setup toolbar](#). With the [persistence options](#) set at their default values, the screen will look something like this:



The colors indicate the frequency of the data. Red is used for the highest-frequency data, with yellow for intermediate frequencies and blue for the least frequent data. In the example above, the waveform spends most of its time in the red region, but noise causes it to wander occasionally into the blue and yellow regions. These are the default colors, but you can change them using the [Persistence Options dialog](#).

This example shows persistence mode in its most basic form. See the [Persistence Options dialog](#) for ways to modify the display to suit your application, and [How to find a glitch using persistence mode](#) for a worked example.

5.13 Measurements table

A **measurements table** displays the results of automatic measurements. Each [view](#) can have its own table, and you can add, delete or edit measurements from this table.

| Channel | Name | Span | Value | Min | Max | Average | σ | Capture Count |
|---------|--------------------|-------------|---------------|---------------|---------------|---------------|---------------|---------------|
| A | AC RMS | Whole trace | 707.7 mV | 707.5 mV | 714.3 mV | 712.5 mV | 2.824 mV | 20 |
| A | Frequency | Whole trace | 4.567 kHz | 3.333 kHz | 7.336 kHz | 5.296 kHz | 997.5 Hz | 20 |
| A | Rise Time [80/20%] | Whole trace | 187.6 μ s | 2.436 μ s | 472.8 μ s | 140.5 μ s | 123.7 μ s | 20 |

| Measurements table columns | |
|----------------------------|---|
| Name | The name of the measurement that you selected in the Add Measurement or Edit Measurement dialog. |
| Span | The section of the waveform or spectrum that you want to measure. This is 'Whole trace' by default. |
| Value | The live value of the measurement, from the latest capture |
| Min | The minimum value of the measurement since measuring began |
| Max | The maximum value of the measurement since measuring began |
| Average | The arithmetic mean of the measurements from the last n captures, where n is set in the General page of the Preferences dialog |
| σ | The standard deviation of the measurements from the last n captures, where n is set in the General page of the Preferences dialog |
| Capture Count | The number of captures used to create the statistics above. This starts at 0 when triggering is enabled, and counts up to the number of captures specified in the General page of the Preferences dialog. |

To add, edit or delete measurements

See: [Measurements toolbar](#).

To change the width of a measurement column

First ensure that the **Column Auto-width** option is not enabled in the [Measurements](#) menu. If necessary, click the option to switch it off. Then drag the vertical separator between column headings to resize the columns, as shown opposite.

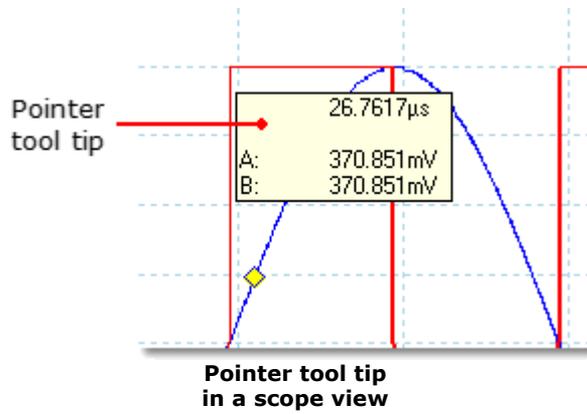
| | Max | Average | |
|---|-----------|----------|---|
| V | 705.8 mV | 705.5 mV | 1 |
| s | 1 ms | 1 ms | 2 |
| V | -8.829 mV | -9.39 mV | 3 |

To change the update rate of the statistics

The statistics (**Min**, **Max**, **Average**, **Standard Deviation**) are based on the number of captures shown in the **Capture Count** column. You can change the maximum capture count using the **Capture Size** control in the [General page](#) of the [Preferences](#) dialog.

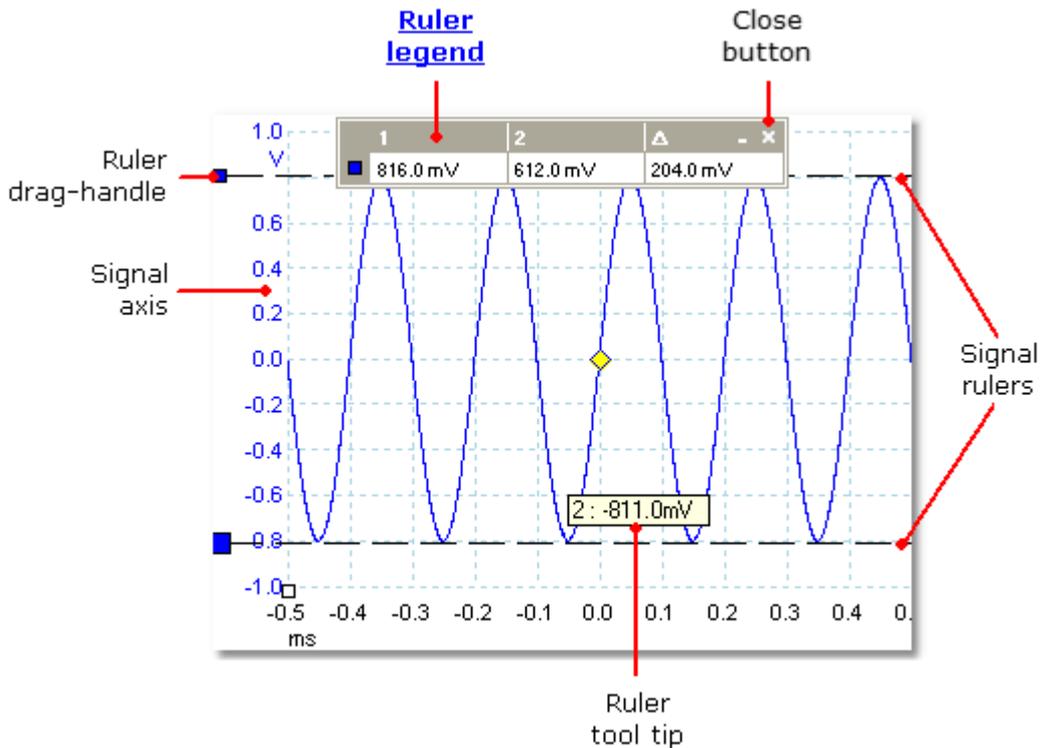
5.14 Pointer tool tip

The **pointer tool tip** is a box that displays the horizontal and vertical axis values at the mouse pointer location. It appears temporarily when you click the background of a [view](#).



5.15 Signal rulers

The **signal rulers** (sometimes called **cursors**) help you measure absolute and relative signal levels on a [scope](#), [XY](#) or [spectrum view](#).



In the [scope view](#) above, the two colored squares to the left of the vertical axis are the **ruler drag-handles** for channel A. Drag one of these downwards from its resting position in the top left corner, and a **signal ruler** (a horizontal dashed line) will extend from it.

Whenever one or more signal ruler is in use, the [ruler legend](#) appears. This is a table showing all of the signal ruler values. If you close the ruler legend using the **Close** button, all the rulers are deleted.

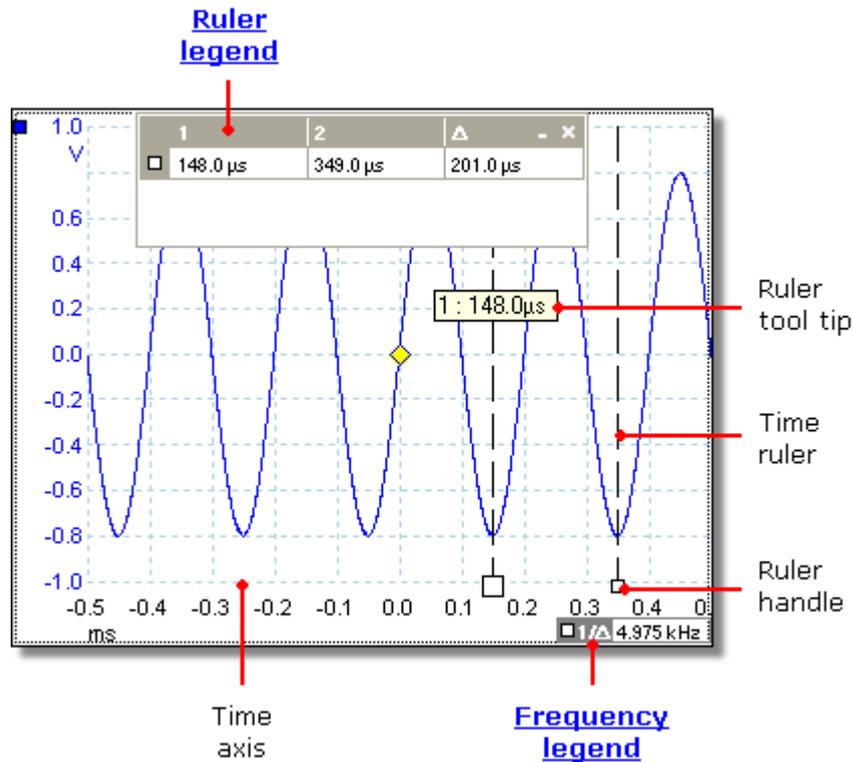
Signal rulers also work in [spectrum](#) and [XY](#) views.

Ruler tool tip

If you move the mouse pointer over one of the rulers, PicoScope displays a [tool tip](#) with the ruler number and the signal level of the ruler. You can see an example of this in the picture above.

5.16 Time rulers

The **time rulers** measure time on a [scope view](#) or frequency on a [spectrum view](#).



In the [scope view](#) above, the two white squares on the time axis are the **time ruler handles**. When you drag these to the right from the bottom left corner, vertical dashed lines called **time rulers** appear. The rulers work in the same way on a [spectrum view](#), but the ruler legend shows their horizontal positions in units of frequency rather than time.

Ruler tool tip

If you hold the mouse pointer over one of the rulers, as we did in the example above, PicoScope displays a tool tip with the ruler number and the time value of the ruler.

Ruler legend

The table at the top of the view is the [ruler legend](#). In this example, the table shows that time ruler 1 is at 148.0 microseconds, ruler 2 is at 349.0 microseconds and the difference between them is 201.0 microseconds. Clicking the **Close** button on the ruler legend also deletes all the rulers.

Frequency legend

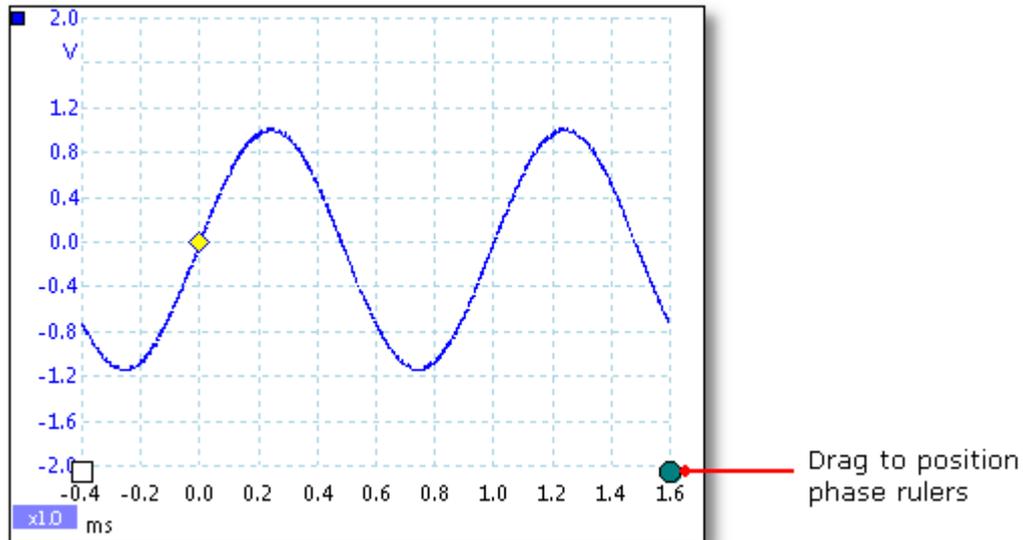
The **frequency legend** in the bottom right-hand corner of a scope view shows $1/\Delta$, where Δ is the difference between the two time rulers. The accuracy of this calculation depends on the accuracy with which you have positioned the rulers. For greater accuracy with periodic signals, use the [frequency measurement](#) function built in to PicoScope.

5.17 Phase (rotation) rulers

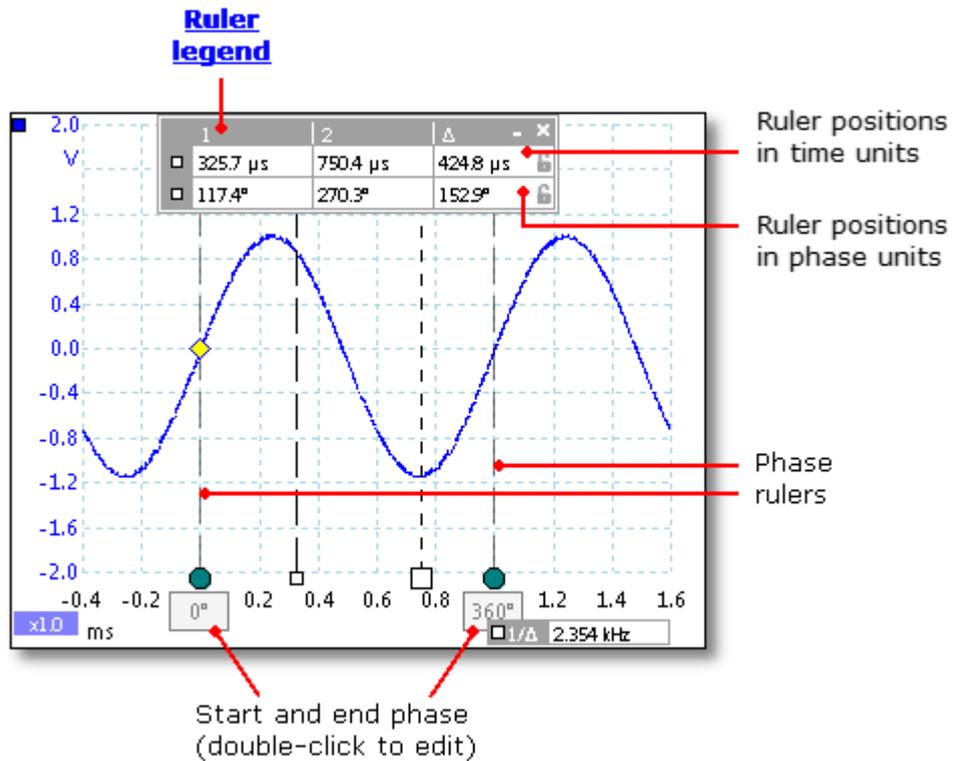
Location: [Scope view](#)

The **phase rulers** (called **rotation rulers** in PicoScope Automotive) help to measure the timing of a cyclic waveform on a [scope view](#). Instead of measuring relative to the trigger point, as [time rulers](#) do, phase rulers measure relative to the start and end of a time interval that you specify. Measurements may be shown in degrees, percent or a custom unit as selected by the [Ruler settings](#) box.

To use the phase rulers, drag the two phase ruler handles onto the waveform from their inactive position as shown below:



When you have dragged both phase rulers into position, the scope view will look like this (we also added two [time rulers](#), for a reason that will be explained later):



In the [scope view](#) above, the two phase rulers have been dragged into place to mark the start and end of a cycle.

The default start and end phase values of 0° and 360° are shown below the rulers and can be edited to any custom value. For example, when measuring timings on a four-stroke engine, it is customary to show the end phase as 720° as one cycle comprises two rotations of the crankshaft.

Ruler legend

The phase rulers become more powerful when used in conjunction with [time rulers](#). When both types of rulers are used together, as shown above, the [ruler legend](#) displays the positions of the time rulers in phase units as well as time units. If two time rulers are positioned, the legend also shows the phase difference between them. Closing the ruler legend dismisses all rulers including the phase rulers.

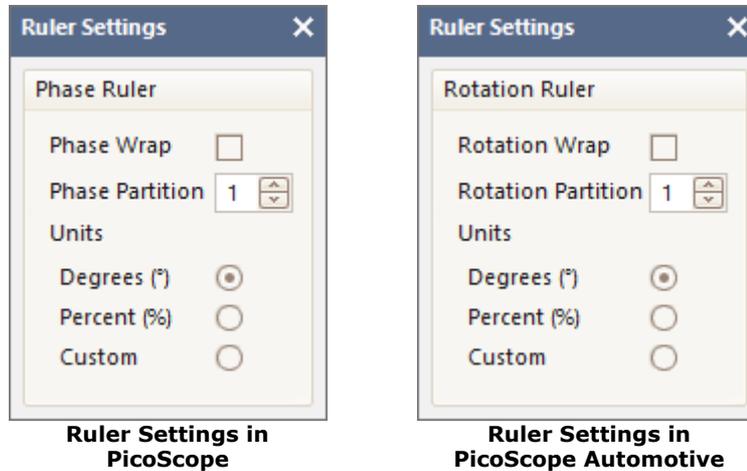
Ruler options

Options for the **phase (rotation) rulers** are configured by the [Ruler Settings dialog](#), which is called up by the **Rulers button** on the [Advanced Options toolbar](#).

5.18 Ruler settings

Location: [Advanced Options toolbar](#) > **Rulers**

The **Ruler Settings** box allows you to control the behavior of the [time rulers](#) and [phase rulers](#) (called **rotation rulers** in PicoScope Automotive).

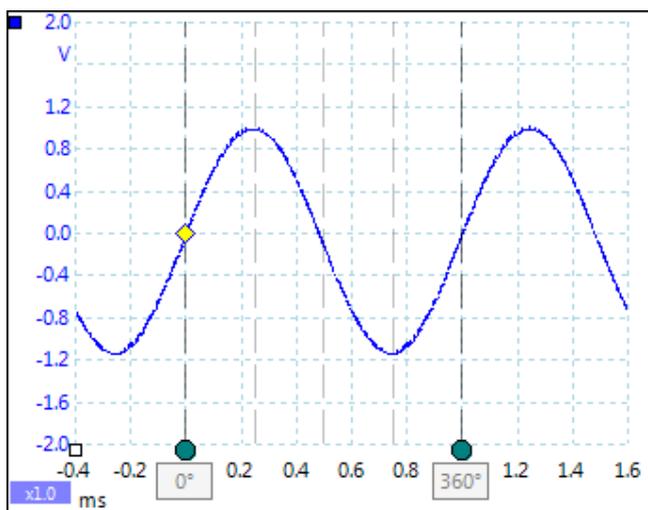


Phase (Rotation) Wrap

If this box is checked, [time ruler](#) values outside the range set by the [phase \(rotation\) rulers](#) are wrapped back into that range. For example, if the phase (rotation) rulers are set to 0° and 360° , the value of a time ruler just to the right of the 360° phase (rotation) ruler will be 0° , and the value of a time ruler just to the left of the 0° phase (rotation) ruler will be 359° . If this box is unchecked, ruler values are unconstrained.

Phase (Rotation) Partition

Increasing this value above 1 causes the space between the two phase (rotation) rulers to be partitioned equally into the specified number of intervals. The intervals are marked by broken lines between the phase (rotation) rulers. The lines help you to interpret complex waveforms such as the vacuum pressure of a four-stroke engine with its intake, compression, ignition and exhaust phases, or a commutated AC waveform in a switch mode power supply.



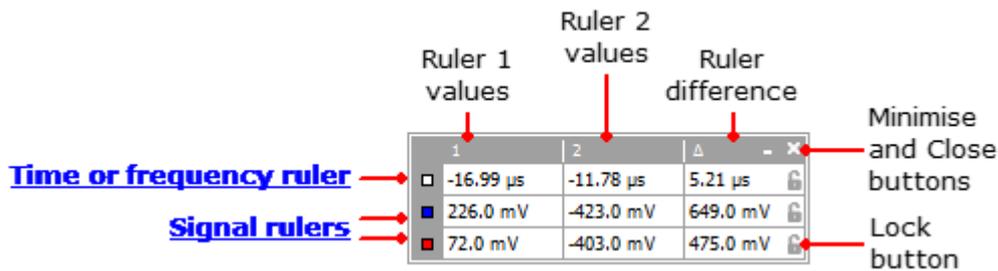
Phase (rotation) rulers with 4 partitions

Units

You can choose between **Degrees**, **Percent** or **Custom**. **Custom** allows you to enter your own unit symbol or name.

5.19 Ruler legend

The **ruler legend** is a box that displays the positions of all the [rulers](#) you have placed on the [view](#). It appears automatically whenever you position a ruler on the view:



Editing

You can adjust the position of a ruler by editing any value in the first two columns. To insert a Greek μ (the *micro* symbol, meaning one millionth or $\times 10^{-6}$), type the letter **u**.

Tracking rulers

When two rulers have been positioned on one channel, the **Lock button**  appears next to that ruler in the ruler legend. Clicking this button causes the two rulers to track each other: dragging one causes the other one to follow it, maintaining a fixed separation. The button changes to  when the rulers are locked.

TIP: To set up a pair of tracking rulers with a known distance between them, first click the **Lock button**, then edit the two values in the ruler legend so that the rulers are the desired distance apart.

Phase (rotation) rulers

When [phase rulers](#) (called **rotation rulers** in PicoScope Automotive) are in use, the ruler legend displays additional information.

See also: [frequency legend](#).

5.20 Frequency legend

 33.37 Hz, 2002.0 RPM

The **frequency legend** appears when you have placed two [time rulers](#) on a [scope view](#). It shows $1/\Delta$ in hertz (the SI unit of frequency, equal to cycles per second), where Δ is the time difference between the two rulers. You can use this to estimate the frequency of a periodic waveform, but you will get more accurate results by creating a frequency measurement using the **Add Measurements button** on the [Measurements toolbar](#).

For frequencies up to 1.666 kHz, the frequency legend can also show the frequency in RPM (revolutions per minute). The RPM display can be enabled or disabled in the [Preferences > Options dialog](#).

5.21 Properties sheet

Location: [Views](#) > **View Properties**

Purpose: shows a summary of the settings that PicoScope 6 is using

The **Properties sheet** appears on the right-hand side of the PicoScope window.

| Properties | | |
|---|-----------------|------------|
| Sampling settings | Sample interval | 64 ns |
| | Sample rate | 15.63 MS/s |
| | No. samples | 781,250 |
| | H/W Resolution | 12 bits |
| Spectrum settings | Window | Blackman |
| | No. bins | 16384 |
| | Bin width | 476.8 Hz |
| | Time gate | 2.097 ms |
| Channel settings | Channel | A |
| | Range | ±10 mV |
| | Coupling | DC |
| | Res-Enhancement | 13.0 Bits |
| | Effective Res | 11 Bits |
| Signal generator settings | Signal type | Square |
| | Frequency | 1 kHz |
| | Amplitude | 1 V |
| | Offset | 0 V |
| Time stamp | Capture Date | 3/5/13 |
| | Capture Time | 12:16:37 |
| Capture rate | Capture Rate | 14 |

No. samples. The number of samples captured. This may be lower than the number requested in the [Maximum Samples](#) control. A number in brackets is the number of interpolated samples if [interpolation](#) is enabled.

Window. The [window function](#) applied to the data before computing the spectrum. This is selected in the [Spectrum options dialog](#).

Time gate. The number of samples that PicoScope uses to compute a spectrum is equal to twice the number of bins. This number of samples is expressed as a time interval called the time gate. It is measured from the start of the capture.

Res-Enhancement (resolution enhancement). The number of bits, including [resolution enhancement](#), selected in the [Channel Options dialog](#).

Effective Res (effective resolution; applies to [Flexible Resolution](#) oscilloscopes only). PicoScope tries to use the value specified by the **Hardware Resolution** control in the [Capture Setup toolbar](#), but on some voltage ranges the hardware delivers a lower effective resolution. The available resolutions are specified in the data sheet for the scope device.

Capture rate. The number of waveforms being captured per second. Shown only in [Persistence Mode](#).

5.22 Custom probes

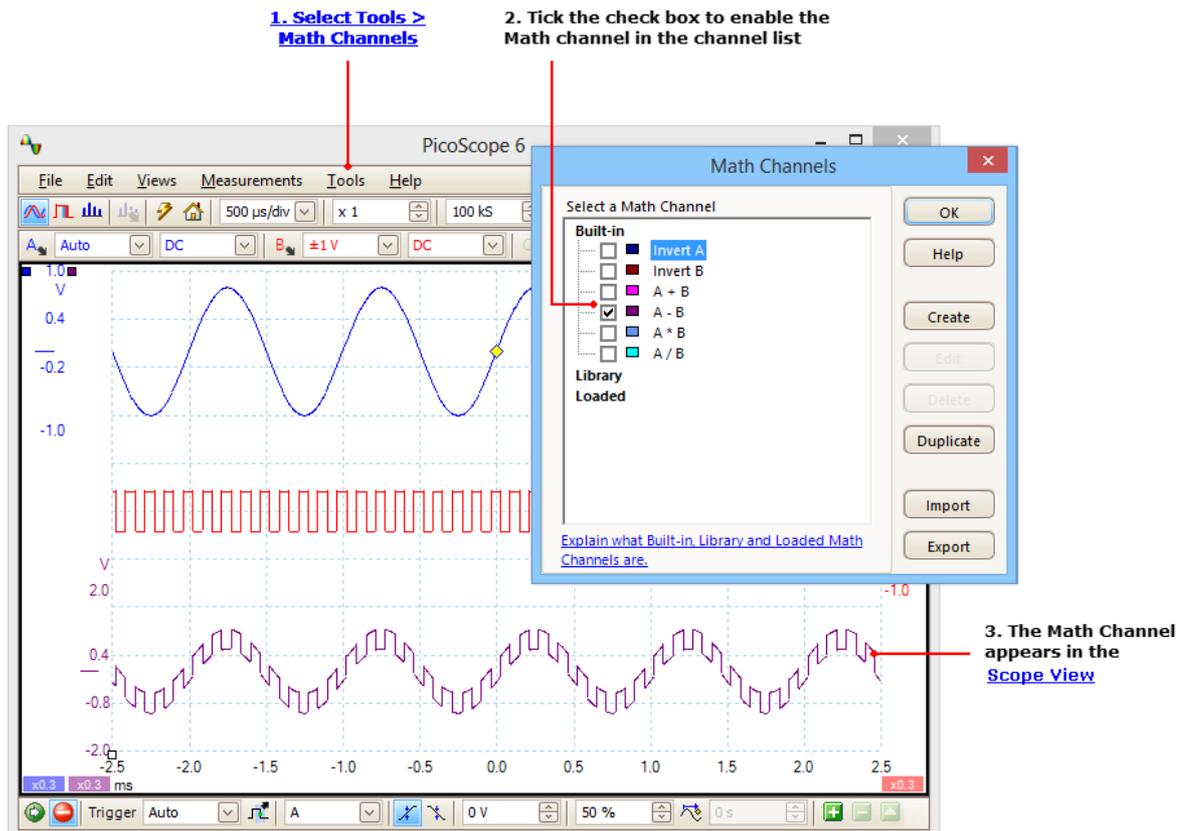
A **probe** is any transducer, measuring device or other accessory that you connect to an input channel of your [scope device](#). PicoScope has a built-in library of common probe types, such as the x1 and x10 voltage probes used with most oscilloscopes, but if your probe is not included in this list you can use the [Custom Probes dialog](#) to define a new one. Custom probes can have any voltage range within the capabilities of the oscilloscope, display in any units, and have either linear or nonlinear characteristics.

Custom probe definitions are particularly useful when you wish to display the probe's output in units other than volts, or to apply linear or nonlinear corrections to the data.

5.23 Math channels

A **math channel** is a mathematical function of one or more input signals. It can be displayed in a [scope](#), [XY](#) or [spectrum](#) view in the same way as an input signal, and like an input signal it has its own measurement axis, [scaling and offset button](#) and [color](#). PicoScope 6 has a set of built-in math functions for the most important functions, such as *Invert A*, *A+B* and *A-B*. You can also define your own functions using the [equation editor](#), or [load predefined math channels from files](#).

Here is a three-step guide to using math channels:



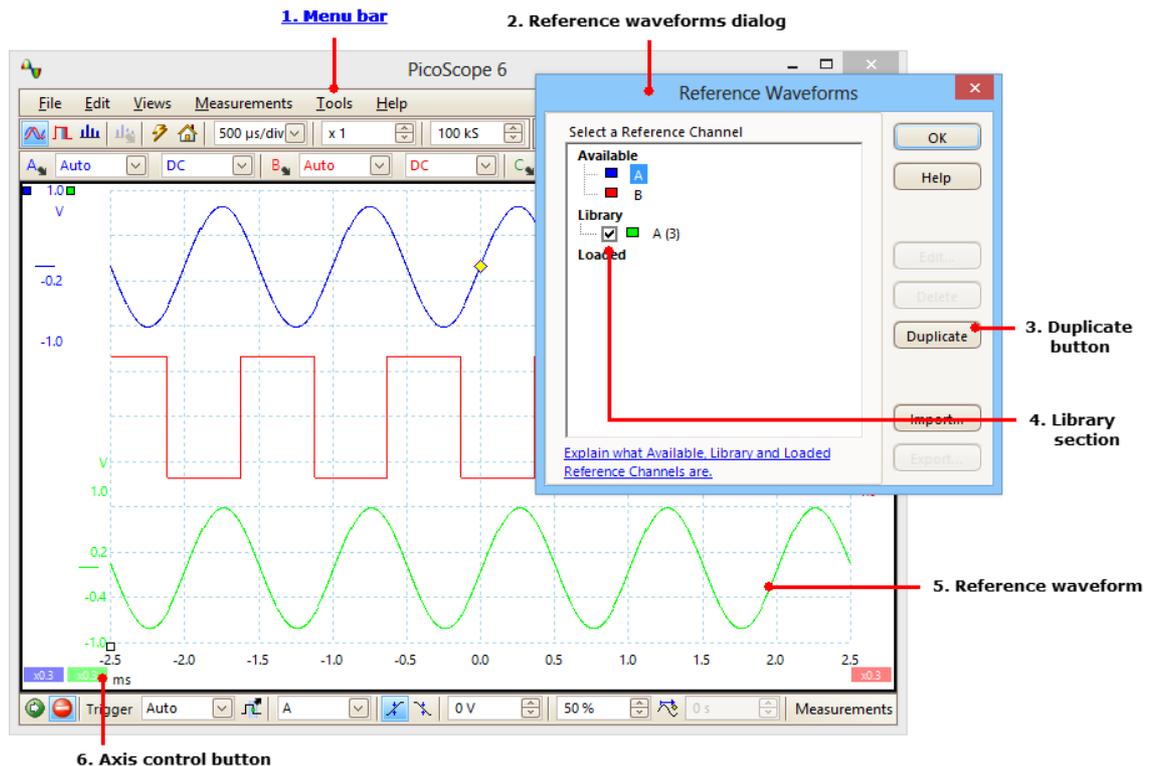
- 1. Tools > Math Channels command.** Click this to open the [Math Channels dialog](#), shown at top right in the picture above.
- 2. Math Channels dialog.** This lists all the available math channels. In the example above, only the built-in functions are listed.
- 3. Math channel.** Once enabled, a math channel appears in the selected [scope](#) or [spectrum](#) view. You can [change its scale and offset](#) as with any other channel. In the example above, the new math channel (bottom) is defined as **A-B**, the difference between input channels A (top) and B (middle).

You may occasionally see a flashing warning triangle -  - at the bottom of the math channel axis. This means that the channel cannot be displayed because an input source is missing. For example, this occurs if you enable the **A+B** function while channel B is set to **Off**.

5.24 Reference waveforms

A **reference waveform** is a stored version of an input signal. You can create one by right-clicking on the view, selecting the **Reference Waveforms** command and selecting which channel to copy. It can be displayed in a scope or spectrum view in the same way as an input signal, and similarly it has its own measurement axis, [scaling and offset button](#), and [color](#). The reference waveform may have fewer samples than the original.

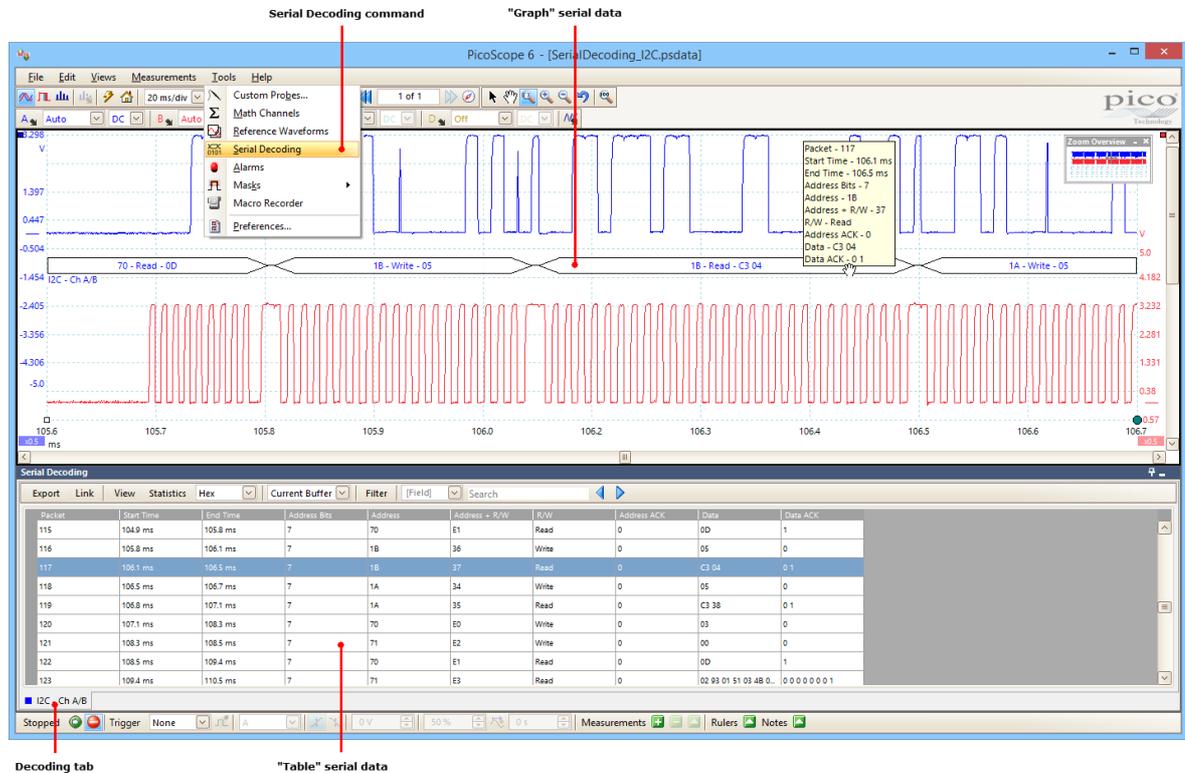
For more control over Reference Waveforms, use the [Reference Waveforms dialog](#) as shown below.



- 1. Reference Waveforms.** Click Tools > Reference Waveforms from the menu bar to open the [Reference Waveforms dialog](#), shown on the right in the picture above.
- 2. Reference Waveforms dialog.** This lists all the available input channels and reference waveforms. In the example above, input channels **A** and **B** are switched on, so they appear in the **Available** section. The **Library** section is empty to begin with.
- 3. Duplicate button.** When you select an input channel or reference waveform and click this button, the selected item is copied to the **Library** section.
- 4. Library section.** This shows all your reference waveforms. Each one has a check box that controls whether or not the waveform appears on the display.
- 5. Reference waveform.** Once enabled, a reference waveform appears in the selected [scope](#) or [spectrum](#) view. You can [change its scale and offset](#) as with any other channel. In the example above, the new reference waveform (bottom) is a copy of channel **A**.
- 6. Axis control button.** Opens an [axis scaling dialog](#) allowing you to adjust scale, offset and delay for this waveform.

5.25 Serial decoding

You can use PicoScope to decode data from a serial bus such as I²C or CAN Bus. Unlike a conventional bus analyzer, PicoScope lets you see the high-resolution electrical waveform at the same time as the data. The data is integrated into the scope view, so there's no need to learn a new screen layout.



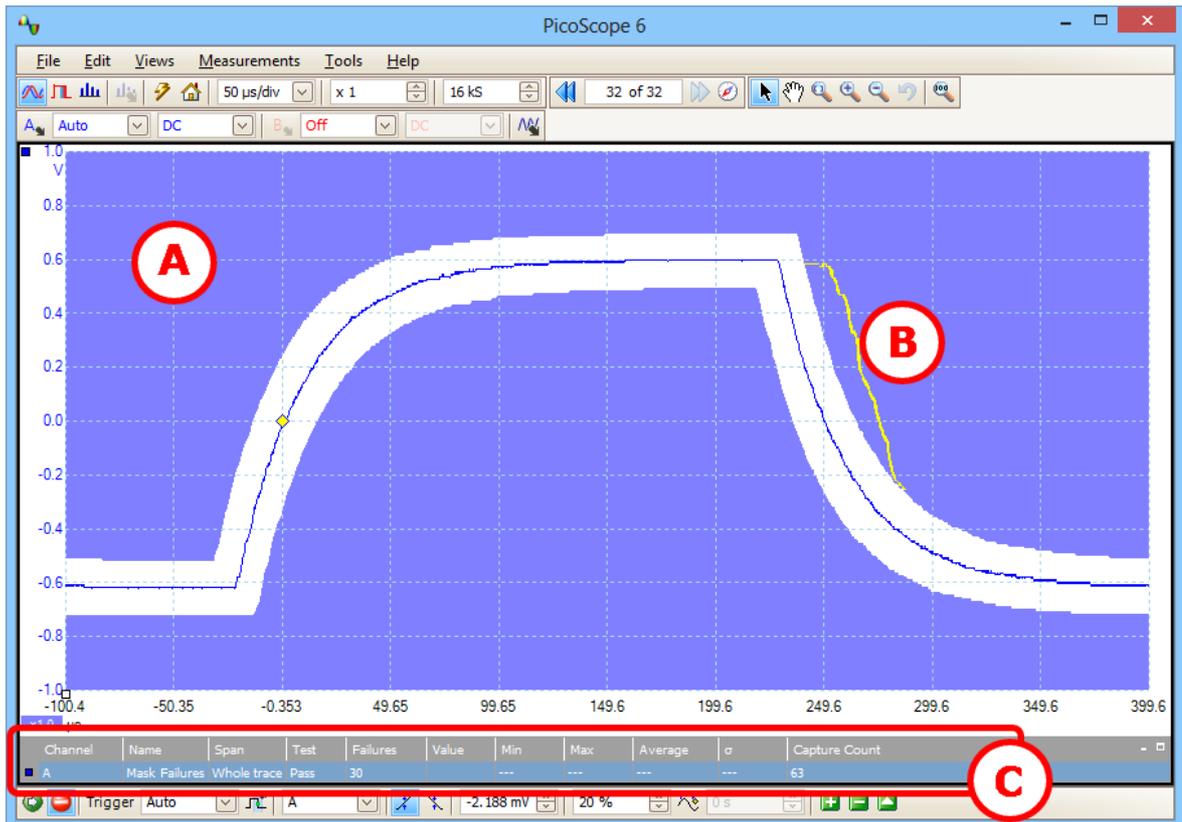
To start decoding your data, click **Tools > Serial Decoding** from the Menu bar. There are many different protocols available, and the data can be viewed in **Graph** or **Table** formats, or both. You can simultaneously decode multiple channels in different formats, and switch between them using the **Decoding tabs**.

For more information, see [Serial Decoding dialog](#).

5.26 Mask limit testing

Mask limit testing is a feature that tells you when a waveform or spectrum goes outside a specified area, called a **mask**, drawn on the [scope view](#) or [spectrum view](#). PicoScope can draw the mask automatically by tracing a captured waveform, or you can draw it manually. Mask limit testing is useful for spotting intermittent errors during debugging, and for finding faulty units during production testing.

To begin, go to the main PicoScope menu and select **Tools > Masks > Add Masks**. This opens the [Mask Library dialog](#). When you have selected, loaded or created a mask, the scope view will appear as follows:



(A) Mask

Shows the allowed area (in white) and the disallowed area (in blue). Right-clicking the mask area and selecting the **Edit Mask** command takes you to the [Edit Mask dialog](#). You can change the mask colors with the **Tools > Preferences > Colors** dialog; add, remove and save masks using the [Masks menu](#); and hide and display masks using the [Views > Masks menu](#).

(B) Failed waveforms

If the waveform enters the disallowed area, it is counted as a failure. The part of the waveform that caused the failure is highlighted, and persists on the display until the capture is restarted.

(C) Measurements table

The number of failures since the start of the current scope run is shown in the [Measurements table](#). You can clear the failure count by stopping and restarting the capture using the [Start/Stop button](#). The measurements table can display [other measurements](#) at the same time as the mask failure count.

5.27 Alarms

Alarms are actions that PicoScope can be programmed to execute when certain events occur. Use the **Tools > Alarms** command to open the [Alarms dialog](#), which configures this function.

The events that can trigger an alarm are:

- Capture - when the oscilloscope has [captured](#) a complete waveform or block of waveforms.
- Buffers Full - when the [waveform buffer](#) becomes full.
- Mask(s) Fail - when a waveform fails a [mask limit test](#).

The actions that PicoScope can execute are:

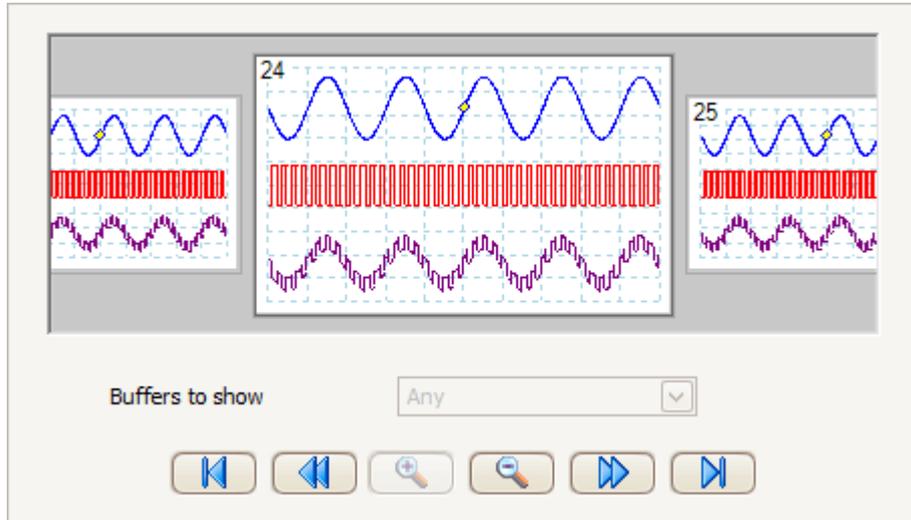
- Beep
- Play Sound
- Stop Capture
- Restart Capture
- Run Executable
- Save Current Buffer
- Save All Buffers

See [Alarms dialog](#) for more details.

5.28 Buffer Overview

The PicoScope waveform buffer can hold up to 10,000 waveforms, subject to the amount of available memory in the oscilloscope. The **Buffer Overview** helps you to scroll through the buffer quickly to find the waveform you want.

To begin, click the **Buffer Overview**  button in the [Buffer Navigation toolbar](#). This opens the **Buffer Overview** window:



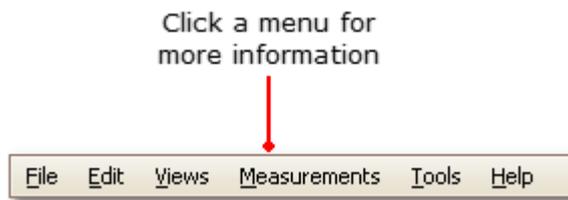
Click on any one of the visible waveforms to bring it to the front of the overview for closer inspection, or use the controls:

- | | |
|--|--|
| Buffers to show | If any of the channels has a mask applied, then you can select the channel from this list. The Buffer Overview will then show only the waveforms that failed the mask test on that channel. |
|  Start: | Scroll to waveform number 1. |
|  Backward: | Scroll to the next waveform on the left. |
|  Zoom in: | Change the scale of the waveforms in the Buffer Overview view. There are three zoom levels: Large: default view. One waveform fills the height of the window. |
|  Zoom out: | Medium: a medium-sized waveform above a row of small waveforms. Small: a grid of small waveforms. Click on the top or bottom row of images to scroll the grid up or down. |
|  Forward: | Scroll to the next waveform on the right. |
|  End: | Scroll to the last waveform in the buffer. (The number of waveforms depends on the Tools > Preferences > General > Maximum Waveforms setting and on the type of scope connected). |

Click anywhere on the main PicoScope window to close the **Buffer Overview** window.

6 Menus

Menus are the quickest way to get to PicoScope's main features. The **Menu bar** is always present at the top of the PicoScope main window, just below the window's title bar. You can click any of the menu items, or press the **Alt** key and then navigate to the menu using the arrow keys, or press the **Alt** key followed by the underlined letter in one of the menu items.

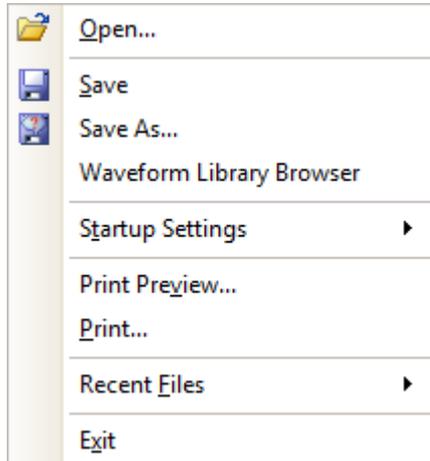


The list of items in the menu bar may vary depending on the windows that you have open in PicoScope.

6.1 File menu

Location: [Menu bar](#) > **File**

Purpose: gives access to file input and output operations



Connect Device. This command appears only when there is no scope device connected, or PicoScope is in Demo mode. It opens the [Connect Device dialog](#), which allows you to select the scope device you wish to use.



Open. Allows you to select the file you want to open. PicoScope can open [.psdata](#) and [.psd](#) files, which contain both waveform data and scope device settings, and [.pssettings](#) and [.pss](#) files, which contain only scope device settings. You can create your own files using the **Save** and **Save As...** commands, described below. If the file was saved using a different scope device from the one that is presently connected, PicoScope may need to modify the saved settings to suit the present device.

Please note that channel information for [.psdata](#) files is displayed in the [Properties sheet](#). If a device is already connected when a [.psdata](#) file is loaded, the channel settings are changed to match the [.psdata](#) file where possible. The resulting settings are displayed on the Channels toolbar.

Hint: Once you have opened a file, you can use the **Page Up** and **Page Down** keys to cycle through all other files in the same directory.



Save. Saves all waveforms using the file name shown in the title bar. If you haven't entered a file name yet, the [Save As dialog](#) opens to prompt you for one.



Save As. Opens the [Save As dialog](#), which allows you to save the settings, waveforms, custom probes and math channels for all views in various formats. Only the waveforms for the mode currently in use ([Scope Mode](#) or [Spectrum Mode](#)) will be saved.

In [persistence mode](#), this command is called **Save Persistence As** and saves only the data for this mode.

Waveform Library Browser. Accesses the [Waveform Library Browser](#).

Startup Settings. Opens the [Startup Settings menu](#).

Print Preview. Opens the **Print Preview window**, which allows you to see how your workspace will be printed when you select the **Print** command.

Print. Opens a standard Windows Print dialog, which allows you to choose a printer, set printing options and then print the selected view.

Recent Files. A list of recently opened or saved files. This list is compiled automatically, but you can clear it using the **Files** page of the [Preferences](#) dialog.

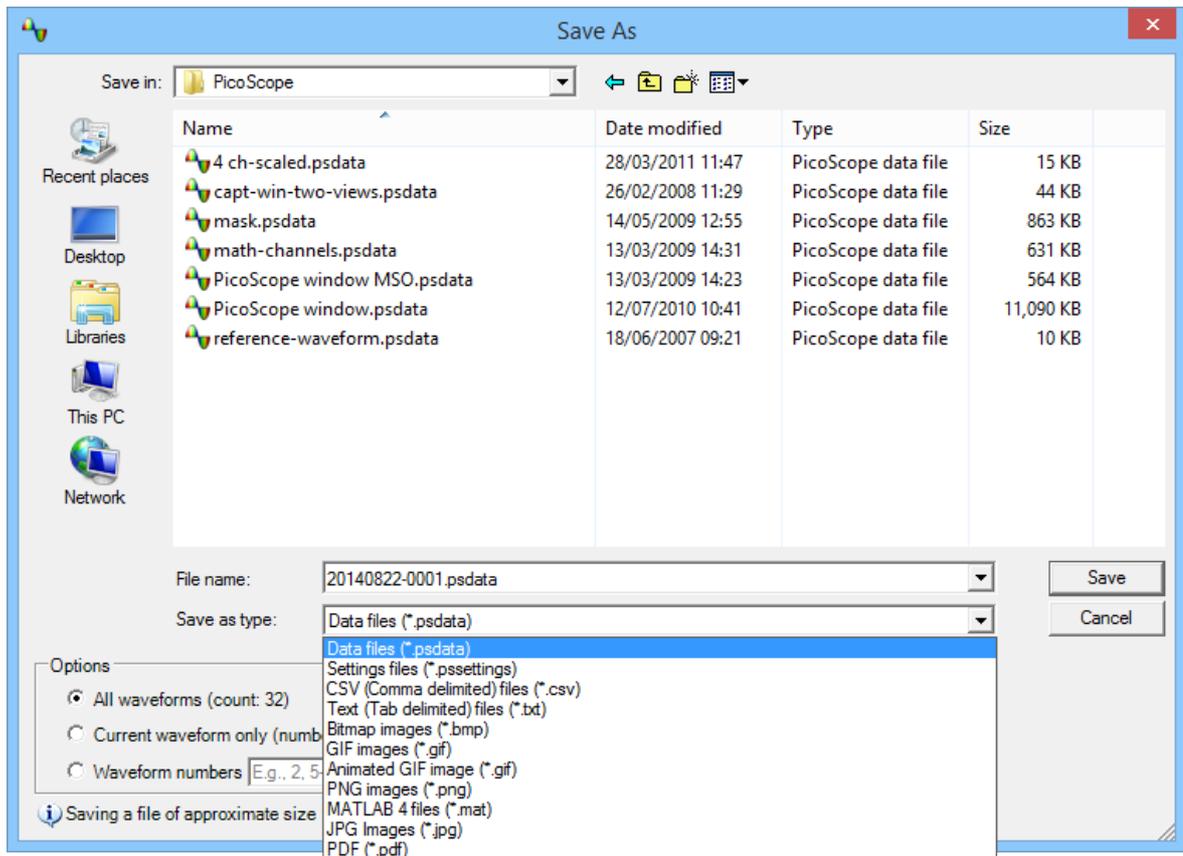
Exit. Close PicoScope without saving any data.

6.1.1 Save As dialog

Location: [File](#) > **Save All Waveforms As** or **Save Current Waveform As**

Purpose: allows you to save your waveforms and settings (including custom probes and active math channels) to a file in [various formats](#)

PicoScope Automotive only: the [Details dialog](#) appears before the **Save As** dialog to allow you to record details of the vehicle and the customer.



Type your chosen file name in the **File name** box, and then select a file format in the **Save as type** box. You can save data in the following formats:

Data files (.psdata)

Stores waveforms and settings from the current scope device. Can be opened on any computer running PicoScope.

| | |
|---|---|
| Settings files (.pssettings) | Stores all settings (but not waveforms) from the current scope device. Can be opened on any computer running PicoScope. |
| CSV (Comma delimited) files (.csv) | Stores waveforms as a text file with comma-separated values. This format is suitable for importing into spreadsheets such as Microsoft Excel. The first value on each line is the time stamp, and it is followed by one value for each active channel, including currently displayed math channels. (Details) |
| Text (Tab delimited) files (.txt) | Stores waveforms as a text file with tab-separated values. The values are the same as those in the CSV format. (Details) |
| Bitmap images (.bmp) | Stores a picture of the waveforms, graticule and rulers in Windows BMP format. The image is 800 pixels wide by 600 pixels high, in 16 million colors, and uncompressed. BMP files are suitable for importing into Windows desktop-publishing programs. |
| GIF images (.gif) | Stores the waveforms, graticule and rulers in CompuServe GIF format. The image is 800 pixels wide by 600 pixels high, in 256 colors, and compressed. GIF files are widely used to illustrate web pages. |
| Animated GIF image (.gif) | Creates an animated GIF that displays all of the waveforms in the buffer in sequence. Each waveform is formatted as in the single GIF format described above. |
| PNG images (.png) | Stores the graticule , rulers and waveforms in PNG format. The image is 800 pixels wide by 600 pixels high, in 16 million colors, with lossless compression. |
| MATLAB 4 files (.mat) | Stores the waveform data in MATLAB 4 format . |
| JPEG (.jpg) | Stores the graticule , rulers and waveforms in JPG format. The image is 800 pixels wide by 600 pixels high, in 16 million colors, with lossy compression. |

Options

The first three options control what happens when the [waveform buffer](#) contains more than one waveform:

All waveform buffers

Save all of the waveforms in the selected file format. If the file format is PSDATA, all of the waveforms are collected in a single file. You can then load this into PicoScope and step through the waveforms using the [buffer navigation controls](#). If the selected file format does not support multiple waveforms, PicoScope creates a new directory containing multiple files.

Current waveform buffer only

Save the single waveform that is currently on view.

Waveform buffers

Save the specified list or range of waveforms. Each waveform is identified by its index number. For example:

```
1,2,9,10  
2, 5-10
```

Only zoomed regions

If the waveform is zoomed horizontally, save only the visible portion.

6.1.1.1 File formats for exported data

PicoScope 6 can export raw data in either text or binary format:

Text-based file formats

- Easy to read without special tools
- Can be imported into standard spreadsheet applications
- Files are very large if there are many samples in the data

[Text file format details](#)

Binary file format

- Files remain relatively small and can even be compressed in some situations (this means that the amount of saved data is unlimited)
- Either a special application is required to read the files or the user must write a program to read the data from the file

If you need to save more than 64 K values per channel, then you must use a binary file format such as the MATLAB® MAT-file format.

[Binary file format details](#)

Data types for storing PicoScope 6 data

Regardless of whether the data types were loaded from a binary file or from a text-based file, we recommend the following data formats for storing the values loaded from a PicoScope 6 data file:

- Sampled data (such as voltages) should use 32-bit single-precision floating-point data types.
- Times should use 64-bit double-precision floating-point data types.

6.1.1.1.1 Text formats

Text-format [files exported by PicoScope 6](#) are encoded in [UTF-8](#) format by default. This is a popular format which is capable of representing a huge range of characters, whilst still retaining some compatibility with the ASCII character set if only standard Western European characters and numbers are used in the file.

CSV (comma-separated values)

CSV files store data in the following format:

```
Time, Channel A, Channel B
(µs), (V), (V)
-500.004, 5.511, 1.215
-500.002, 4.724, 2.130
-500, 5.552, 2.212
...
```

There is a comma after each value on a line to represent a column of data and a carriage return at the end of the line to represent a new row of data.

Note. CSV files are not the best choice of format if you are working in a language that uses the comma character as the decimal point. Instead, try using the tab-delimited format which works in almost the same way.

Tab-delimited

Tab-delimited files store data in the following format:

```
Time          Channel A    Channel B
(µs)          (V)          (V)
500.004      5.511        1.215
-500.002     4.724        2.130
-500         5.552        2.212
...
```

The files have a tab character after each value on a line to represent a column of data and a carriage return at the end of the line to represent a new row of data. These files work in any language and are a good choice for sharing data internationally.

6.1.1.1.2 Binary formats

PicoScope 6 can [export data](#) in version 4 of the .mat binary file format. This is an open format and the full specification is available as PDF documentation from the www.mathworks.com website. PicoScope 6 saves data into the MAT-file in a specific way, which is detailed below.

Note: Data arrays saved in version 4 of the .mat file format must not exceed the maximum size of 100 million elements. MATLAB cannot read arrays with more elements than this in this format.

Importing into MATLAB

Load the file into your workspace using this syntax:

```
load myfile
```

Each channel's data is stored in an array variable named by the channel. So, the sampled data for channels A to D would be in four arrays named **A**, **B**, **C** and **D**.

There is only one set of time data for all channels and this is loaded in one of two possible formats:

1. A start time, an interval and a length. The variables are named **Tstart**, **Tinterval** and **Length**.
2. An array of times (sometimes used for ETS data). The time array is named **T**.

If the times are loaded in as **Tstart**, **Tinterval** and **Length** then you can use the following command to create the equivalent array of times:

```
T = [Tstart : Tinterval : Tstart + (Length - 1) * Tinterval];
```

Note: The size of the largest file that MATLAB can open depends on the resources of the computer. It is therefore possible for PicoScope to create a MATLAB file that some installations of MATLAB would be unable to open. Please be aware of this risk when saving critical data.

The other variables used in the file are:

| | |
|------------------------|---|
| ExtraSamples | the number of samples added, for example when filtering or resolution enhancement is enabled |
| RequestedLength | the number of samples. This will be the same as "No. samples" in the PicoScope 6 Properties sheet . Note that, if the signal is not acquired in its entirety, this will be different from Length . |
| Version | the exact version of PicoScope 6 used to export the file |

Exploring the file format

The full file specification, available from www.mathworks.com, is comprehensive so this guide does not describe the entire format. Instead, this guide describes enough of the format to allow you to get data from the file and use it in your own program.

The variables described above (under [Importing into MATLAB](#)) are stored in a series of data blocks, each preceded by a header. Each variable has its own header and data block and the corresponding variable names are stored with them (such as **A**, **B**, **Tstart**). The following sections describe how to read each variable from the file.

The order of the data blocks is not specified, so programs should look at the variable names to decide which variable is currently being loaded.

● The header

The file consists of a number of data blocks preceded by 20-byte headers. Each header contains five 32-bit integers (as described in the table below).

| Bytes | Value |
|----------|---------------------------|
| 0 to 3 | Data format (0, 10 or 20) |
| 4 to 7 | Number of values |
| 8 to 11 | 1 |
| 12 to 15 | 0 |
| 16 to 19 | Name length |

● Data format

The *Data format* in bytes 0 to 3 describes the type of numerical data in the array.

| Value | Description |
|-------|--------------------------------|
| 0 | Double (64-bit floating point) |
| 10 | Single (32-bit floating point) |
| 20 | Integer (32-bit) |

● Number of values

The *Number of values* is a 32-bit integer describing the number of numerical values in the array. This value may be 1 for variables that only describe one value; but for arrays of samples or times, expect this to be a large number.

● Name length

The *Name length* is the length of the name of the variable as a null-terminated 1-byte per character ASCII string. The last null terminating character (`\0`) is included in the *Name length* so if the variable name is *TStart* (same as *TStart\0*) then the name length will be 7.

● The data block

The data block begins with the name of the variable (such as **A**, **Tinterval**) and you should read in the number of bytes described by the *Name length* part of the header (not forgetting that the last byte in the string is `\0` if your programming language needs to take account of this).

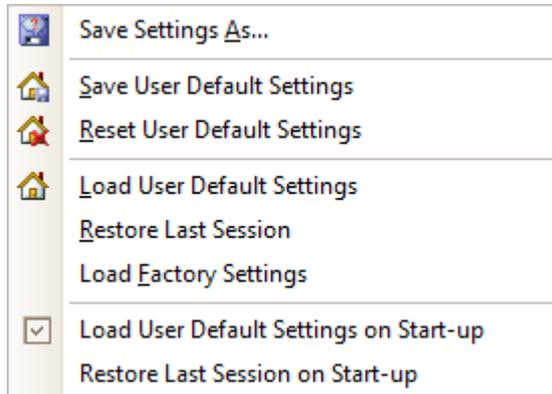
The remaining part of the data block is the actual data itself, so read in the number of values described in the *Number of values* part of the header. Remember to take account of the size of each value as described in the 'Data format' part of the header.

Channel data such as voltages, in variables such as **A** and **B**, are stored as 32-bit single-precision floating-point data types. Times such as **Tstart**, **Tinterval** and **T** are stored as 64-bit double-precision floating-point data types. **Length** is stored as a 32-bit integer.

6.1.2 Startup Settings menu

Location: [File](#) > **Startup Settings**

Purpose: allows you to load, save and restore the PicoScope 6 startup settings



Commands

Save Settings As. Stores all the current settings (for example timebase, voltage range, trigger), which can then be opened on any computer running PicoScope 6. This allows you to create a library of different setups.

Save User Default Settings. Saves the current settings as the default. This means you only have to configure the oscilloscope once in order to use these settings every time you open PicoScope 6.

Reset User Default Settings. This replaces your saved default settings with the original factory settings.

Load User Default Settings. Resets all settings to the saved default. You can also use the home button on the toolbar to achieve this.

Restore Last Session. This option returns the settings to those from your last session.

Load Factory Settings. Returns the settings to the factory original.

Options

Load User Default Settings on Start-up. Selecting this option ensures that the next time you start up PicoScope 6, the saved user default settings will be applied.

Restore Last Session on Start-up. You should enable this checkbox if you want PicoScope 6 to open with the settings you used for your last session.

6.5.1.1.6 Manual Ranges Setup dialog

Location: [Range Management dialog](#) > **Advanced** > **Next**

Purpose: creates ranges manually for your [custom probe](#)

How to use the dialog

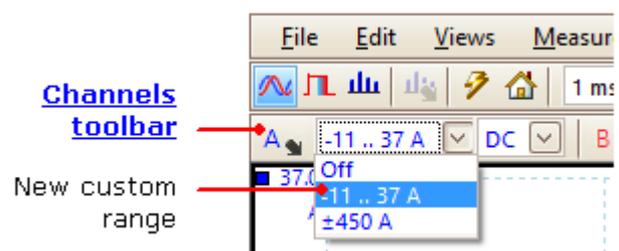
If you wish, you can click **Auto Generate Ranges** and the program will create a number of ranges for the selected device. This will create the same list of ranges that you would have obtained by selecting **Let the software manage my ranges for me automatically** in the previous dialog. When you select a range, a diagram below the list will show its relationship to the scope device's input range — this is explained further under [Edit range dialog](#). You can then edit the ranges by clicking **Edit**, or you can also add a new range by clicking **New Range**. Both of these buttons take you to the [Edit Range dialog](#).

Click **Next** to continue to the [Filter Method dialog](#).

Click **Back** to return to the [Range Management dialog](#).

How to use a new custom range

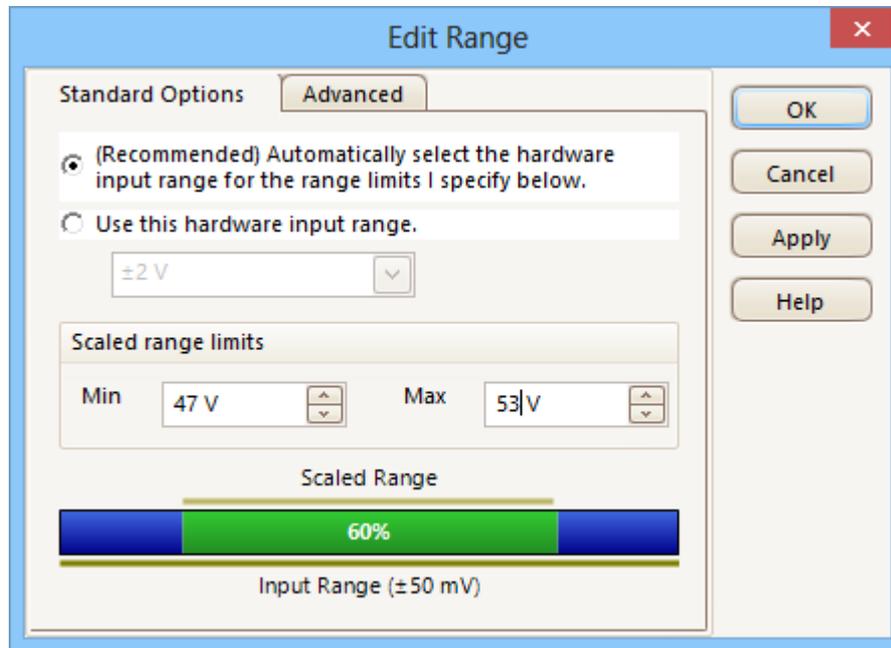
After you have created a custom range, it will appear in the drop-down list of ranges in the [channels toolbar](#), like this:



6.5.1.1.6.1 Edit Range dialog

Location: [Manual Ranges Setup dialog](#) > **Edit** or **New Range**

Purpose: editing a manual range for a [custom probe](#)



Automatic mode

If you leave the **Automatic** radio button pressed, the program will automatically determine the best hardware input range for the device as you change the **Scaled range limits**. This is the best mode to use for almost all ranges. You should set the **Scaled range limits** to the maximum and minimum values you wish to see on the vertical axis of the scope display.

Fixed range mode

If you press the **Hardware input range** radio button and select a hardware input range from the drop-down box, PicoScope will then use that hardware input range whatever scaled range limits you choose. Set the upper and lower scaled range limits to the limits you wish to appear at the top and bottom of the vertical axis in PicoScope's [scope view](#).

What is an input range?

An input range is the signal range, usually in volts, on the input channel of the [scope device](#). Your scaled range should match this as closely as possible to make the most of the scope's resolution.

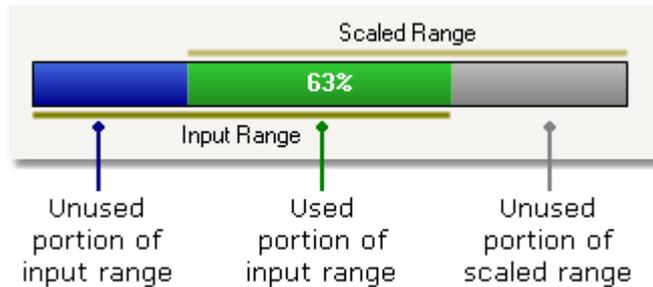
What is a scaled range?

The scaled range is the range that will appear on the vertical axis of the scope display when the probe is selected.

The scaling that you chose on the [Scaling Method](#) page defines the relationship between the input range and the scaled range. This dialog enables you to set up ranges to display the scaled data on the scope view.

The range utilization bar

This diagram at the bottom of the dialog shows how well the input range of the device is matched to the scaled range.



- **Green** - The section of the input range that is used by the scaled range. This should be as large as possible, to maximise the use of the scope device's resolution.
- **Blue** - Areas of the input range that are not being used. These indicate wasted resolution.
- **Grey** - Parts of the scaled range that are not covered by the input range. These will result in wasted space on the graph. The range utilization bar may not represent these areas accurately when non-linear scaling is being used, so you should always test the scaled range limits on the scope view.

[Advanced tab](#)

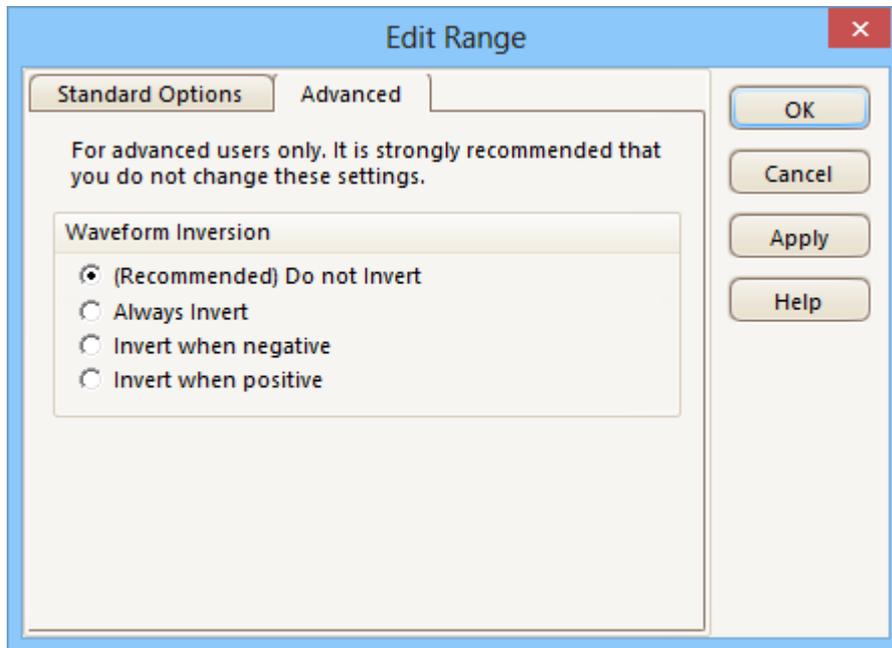
Finishing

Clicking **OK** or **Cancel** will return you to the [Manual Ranges Setup dialog](#).

6.5.1.1.6.2 Edit Range dialog (Advanced tab)

Location: [Manual Ranges Setup dialog](#) > **Edit** or **New Range** > **Advanced** tab

Purpose: configuring advanced options for [custom probes](#)



These options are for factory use and we recommend that you do not change them.

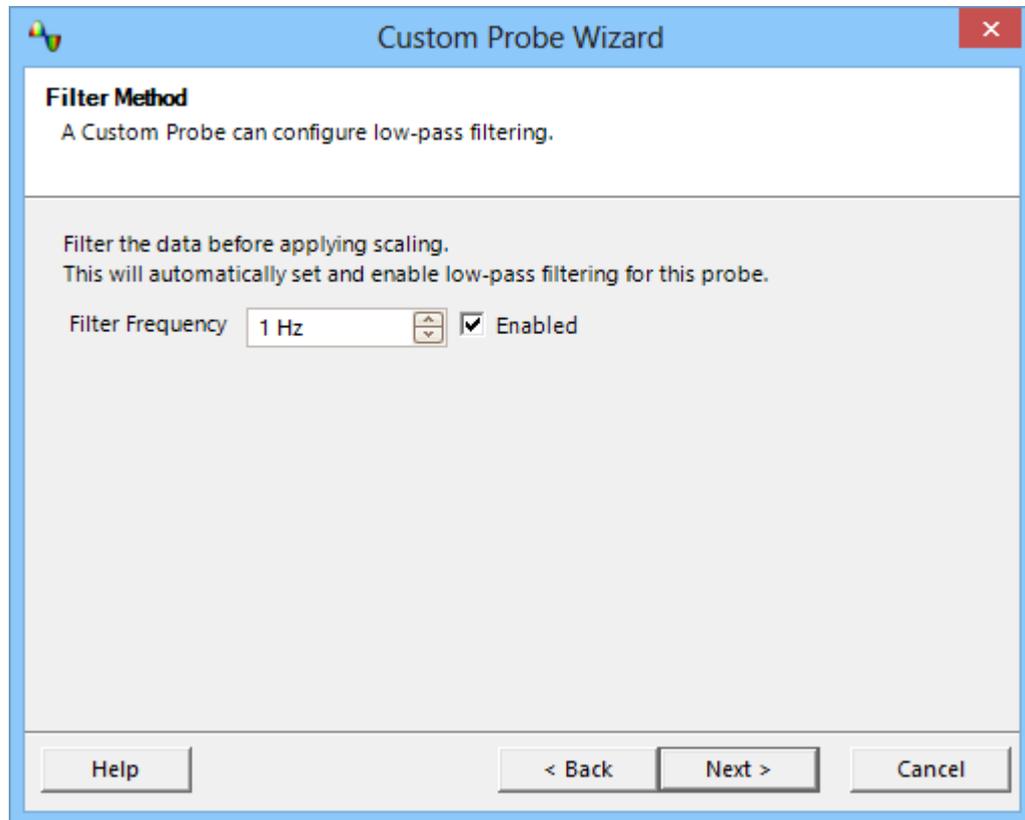
Finishing

Clicking **OK** or **Cancel** will return you to the [Manual Ranges Setup dialog](#).

6.5.1.1.7 Filter Method dialog

Location: [Manual Ranges Setup dialog](#) > **Next**

Purpose: sets up lowpass filtering for this custom probe



This dialog has the same effect as manually enabling the [Lowpass Filtering](#) option in the [Channel Options dialog](#). Filtering will only occur if the attached scope device supports filtering.

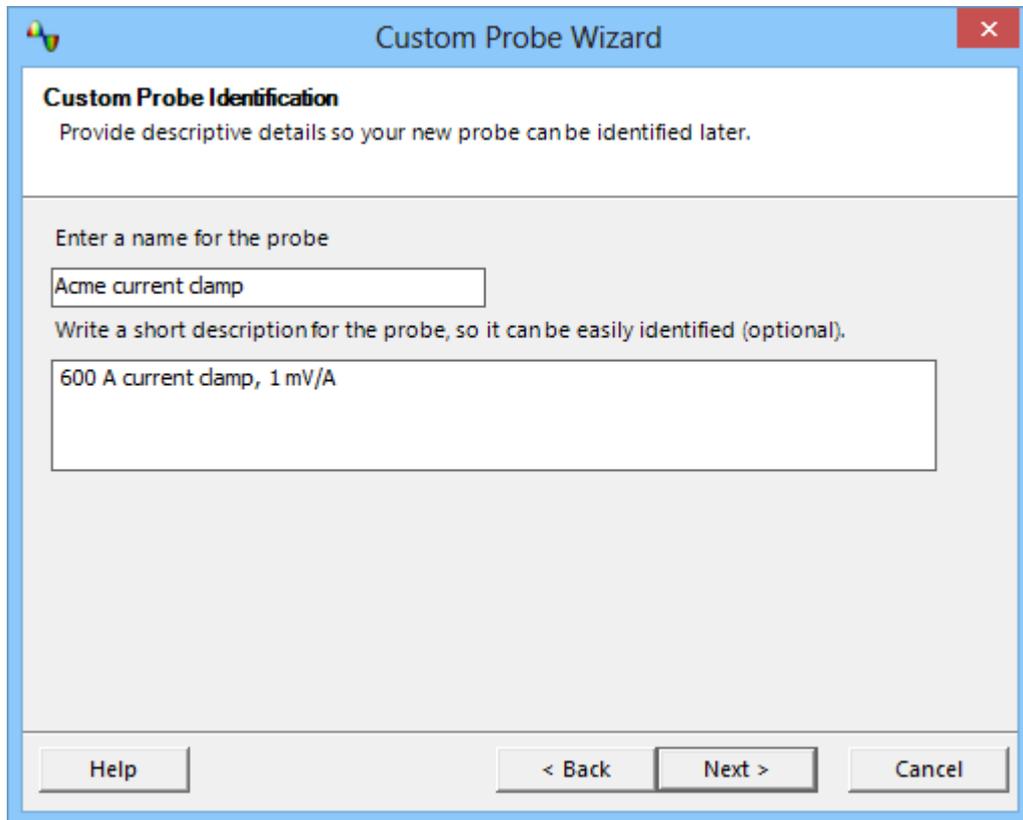
Back: Go to the [Manual Range Setup dialog](#)

Next: Go to the [Custom Probe Identification dialog](#)

6.5.1.1.8 Custom Probe Identification dialog

Location: [Range Management dialog](#) > **Next**

Purpose: entering text to identify the [custom probe](#)



Custom Probe Wizard

Custom Probe Identification
Provide descriptive details so your new probe can be identified later.

Enter a name for the probe

Acme current clamp

Write a short description for the probe, so it can be easily identified (optional).

600 A current clamp, 1 mV/A

Help < Back Next > Cancel

How to use the dialog

Click **Back** to return to the [Filter Method dialog](#).

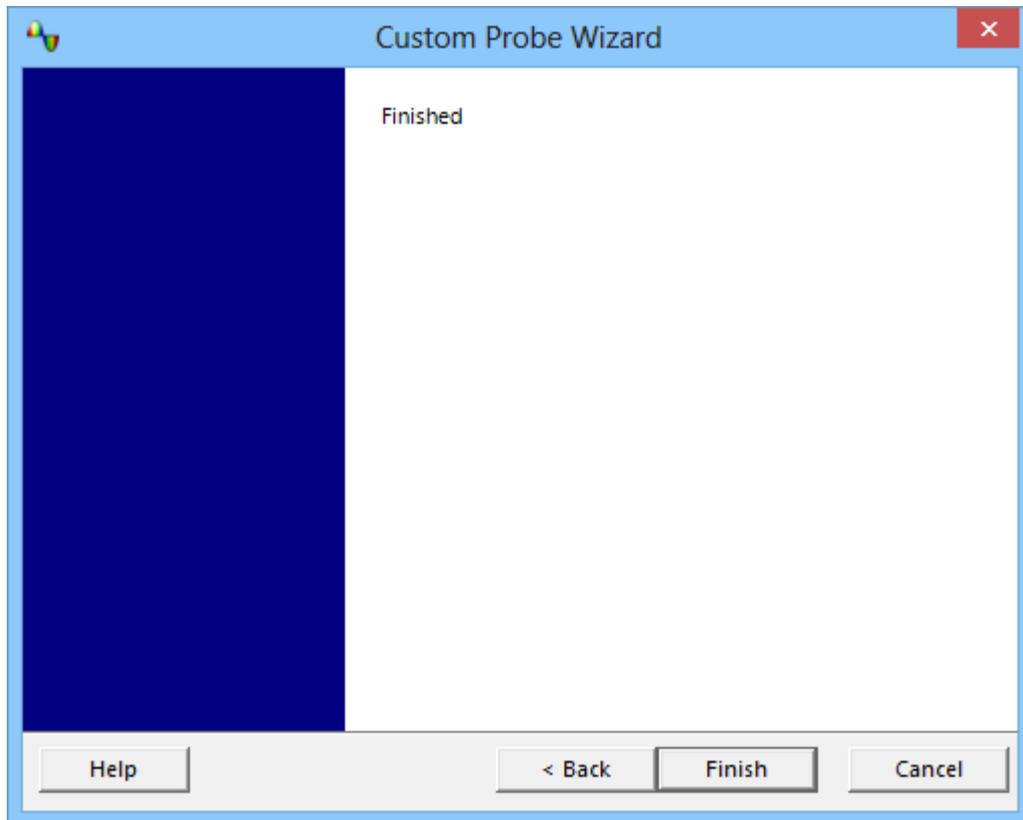
- The **probe name** will appear in the probe list.
- The **description** is not used in the present version of the software.

Fill in the text fields and click **Next** to continue to the [Custom Probe Finished dialog](#).

6.5.1.1.9 Custom Probe Finished dialog

Location: [Custom Probe Identification dialog](#) > **Next**

Purpose: signals the end of the [custom probe](#) setup procedure

**How to use the dialog**

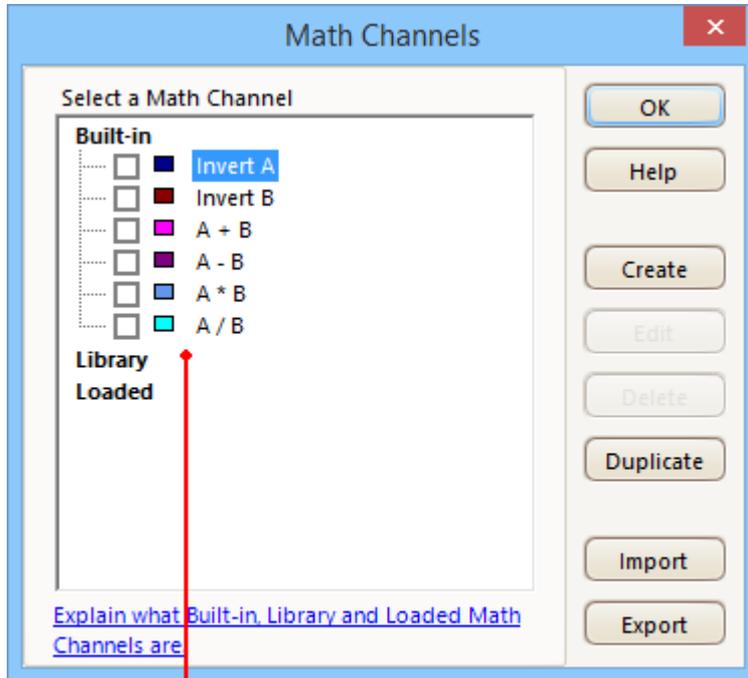
Click **Back** to return to the [Custom Probe Identification dialog](#).

Click **Finish** to accept your custom probe settings and return to the [Custom Probes dialog](#).

6.5.2 Math Channels dialog

Location: [Tools](#) > [Math Channels](#)

Purpose: [creating](#), [editing](#) and controlling [math channels](#), which are virtual channels generated by mathematical functions of input channels



Math Channel list

Math Channel list The main area of the **Math Channels dialog** is the **Math Channel list**, which shows all the built-in, library and loaded [math channels](#). To choose whether or not a channel appears in the main [PicoScope window](#), click the appropriate check box and then **OK**. You can have up to 8 channels in any view, including input channels and math channels. If you enable a 9th channel, PicoScope opens a new view.

Built-in: these math channels are defined by PicoScope and cannot be changed

Library: these are the math channels that you define using the **Create** or **Duplicate** button, **Edit**, or load with the **Import** button

Loaded: these are the math channels present in any PicoScope settings or data files that you have loaded

Create

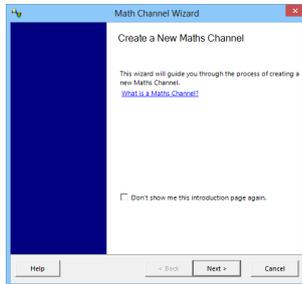
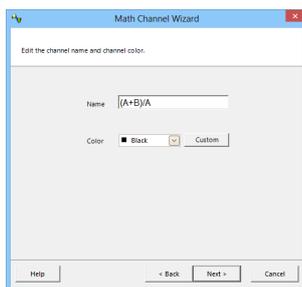
Opens the [Math Channel Wizard](#), which guides you through the process of creating or editing a math channel. The new channel will appear under **Library** in the **Math Channel list**.

| | |
|------------------|---|
| Edit | Opens the Math Channel Wizard to allow you to edit the selected math channel. You must first select a channel in the Library section of the Math Channel list . If the channel you want to edit is in the Built-in or Loaded section, first copy it to the Library section by clicking Duplicate , then select it and click Edit . |
| Delete | Permanently deletes the selected math channel. Only math channels in the Library section can be deleted. |
| Duplicate | Creates a copy of the selected math channel. The copy is placed in the Library section, from where you can edit it by clicking Edit . |
| Import | Opens a <code>.psmaths</code> math channel file and places the math channels it contains in the Library section. |
| Export | Saves all math channels from the Library section to a new <code>.psmaths</code> file. |

6.5.2.1 Math Channel Wizard

Location: [Channel Setup toolbar](#) > **Math Channels button**

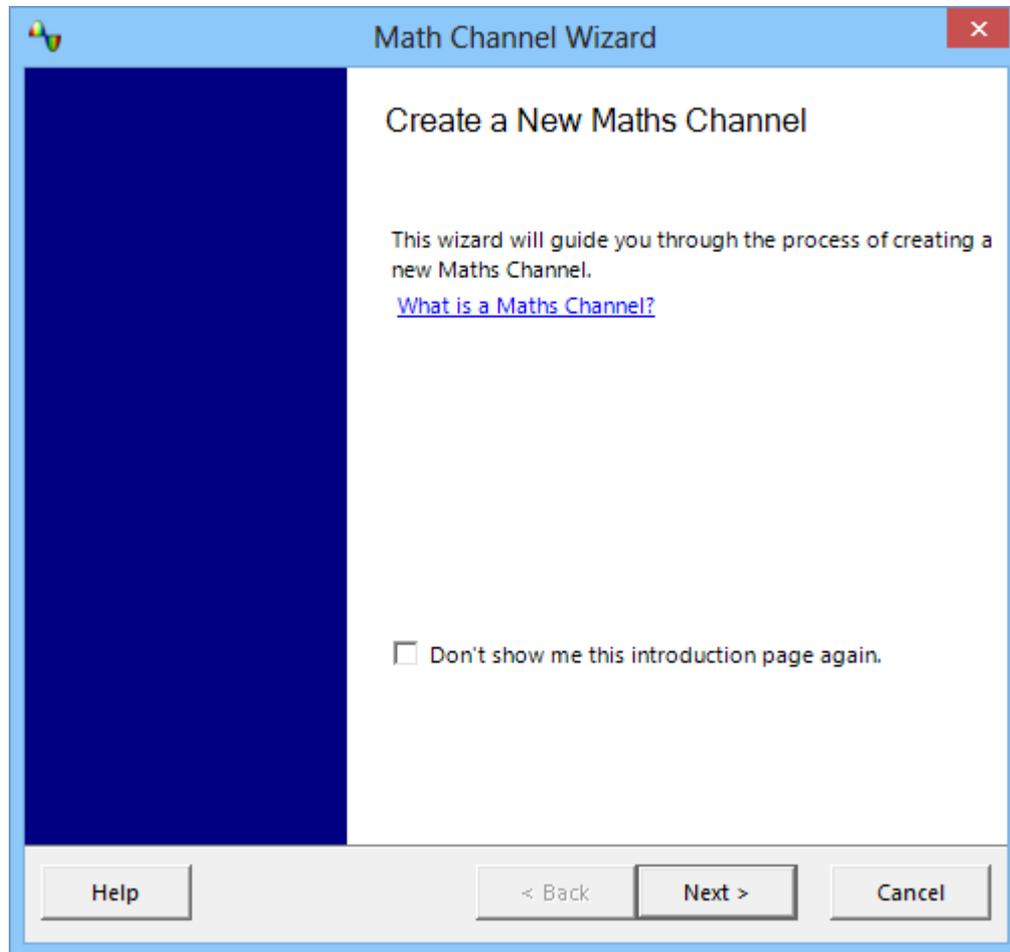
Purpose: creating, editing and controlling math channels, which are virtual channels generated by mathematical functions of input channels

1. [Introduction](#)2. [Equation](#)3. [Channel name](#)4. [Units and range](#)5. [Finished](#)

6.5.2.1.1 Math Channel Wizard Introduction dialog

Location: [Math Channels dialog](#) > **Create** (if you have not checked the *Don't show me this introduction page again* box)

Purpose: introduces the [Math Channel Wizard](#)



6.5.2.1.2 Math Channel Wizard Equation dialog

Location: [Math Channel Wizard](#)

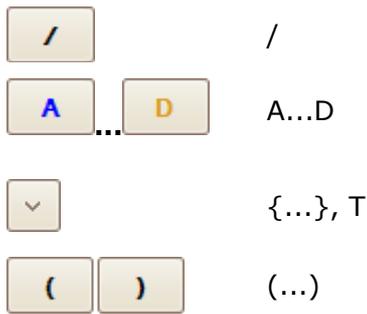
Purpose: allows you to enter or edit the equation for a [math channel](#). You can type directly into the equation box, or click the calculator buttons and let the program insert the symbols for you. A red error indicator  will appear to the right of the equation box if the equation contains a syntax error.

Basic view

Math Channel Wizard Equation dialog, basic view

Basic buttons

| Button | Equation | Description |
|---|----------|--|
|  | | Clear Equation. Clears the entire contents of the equation box. |
|  | | Clear. Clears the single character to the left of the cursor. |
|  | + | Add |
|  | - | Subtract (or negate) |
|  | * | Multiply |



Divide

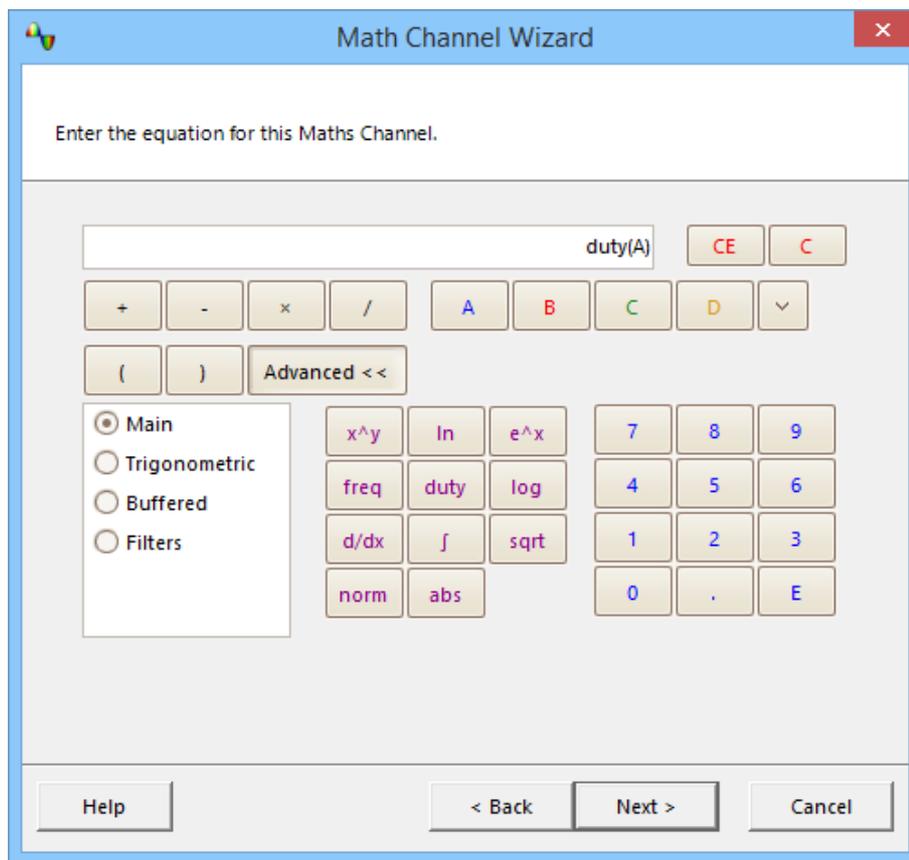
Input Channels. The selection available depends on the number of enabled channels on the scope device.

Other Operands. Shows a drop-down list of available inputs to equations, including [reference waveforms](#) and **Time**.

Brackets. Expressions within brackets will be evaluated before the expressions to either side.

Advanced view

Clicking the **Advanced** button reveals more function buttons. Initially these are the **Main** group of buttons, as shown below.



Equation Editor Advanced view, showing Main buttons

Advanced buttons (numeric keypad)

| Button | Equation | Description |
|--------|----------|---|
| | 0..9 | 0 to 9. The decimal digits. |
| | . | Decimal point |
| | E | Exponent. aEb means $a \times 10^b$. |

Advanced buttons (Main group)

| Button | Equation | Description |
|--|--------------|---|
|  | sqrt() | Square root |
|  | x^y | Power. Raise x to the power of y . |
|  | ln() | Natural logarithm |
|  | abs() | Absolute value |
|  | freq() | Frequency. Calculated in hertz. |
|  | duty() | Duty. Calculate the duty cycle as a percentage. |
|  | norm() | Normalise. PicoScope calculates the maximum and minimum values of the argument over the capture period, and then scales and offsets the argument so that it exactly fits the range [0, +1] units. |
|  | exp() | Natural exponent. Raise e , the base of the natural logarithm, to the power of x . |
|  | log() | Logarithm. Base-10 logarithm. |
|  | derivative() | Derivative. Calculated with respect to the x -axis. Note: the derivative of a sampled signal contains a large amount of noise, so it is advisable to apply digital low-pass filtering to all channels used as inputs to this function. |
|  | integral() | Integral. Along the x -axis. |

Click **Trigonometric**, **Buffered** or **Filters** to reveal alternative groups of buttons:

Advanced buttons (Trigonometric group)

| Button | Equation | Description |
|---|----------|--|
|  | pi | Pi. The ratio of a circle's circumference to its diameter. |
|  | | Inverse. Modifies the sin , cos and tan buttons to the inverse trigonometric functions asin , acos and atan . |
|  | sin() | Sine. The operand is in radians. |
|  | cos() | Cosine. The operand is in radians. |
|  | tan() | Tangent. The operand is in radians. |
|  | sinh() | Hyperbolic sine. |
|  | cosh() | Hyperbolic cosine. |
|  | tanh() | Hyperbolic tangent. |

Advanced buttons (Buffered group)

When the scope is running, these functions operate continuously on all waveforms since the scope started capturing. If a math channel containing these functions is enabled when the scope is stopped, it operates on the contents of the waveform buffer instead.

| Button | Equation | Description |
|---|-----------|---|
|  | min() | Minimum. Negative peak detect of all previous waveforms. |
|  | max() | Maximum. Positive peak detect of all previous waveforms. |
|  | average() | Average. Arithmetic mean of all previous waveforms. |
|  | peak() | Peak detect. Display maximum-to-minimum range of all previous waveforms. |

Advanced buttons (Filters group)

Parameters:

- i is the input channel or other operand (see under **Basic buttons** above)
- f (or f_1 and f_2) are the -3 dB cutoff frequencies of the filter, in hertz

| Button | Equation | Description |
|---|---------------------------|---|
|  | HighPass(i, f) | High pass filter. Attenuates low frequencies. |
|  | LowPass(i, f) | Low pass filter. Attenuates high frequencies. |
|  | BandPass(i, f_1, f_2) | Band pass filter. Attenuates high and low frequencies outside the specified range. |
|  | BandStop(i, f_1, f_2) | Band stop filter. Attenuates mid-band frequencies inside the specified range. |

These are digital filters with a finite number of taps and therefore cannot attenuate down to DC. They have a minimum cutoff frequency of 1/64 000 of the scope's sampling rate. You can find the current sampling rate displayed in the [Properties sheet](#).

Additional functions

There are a few operators that can be entered only by using the equation box:

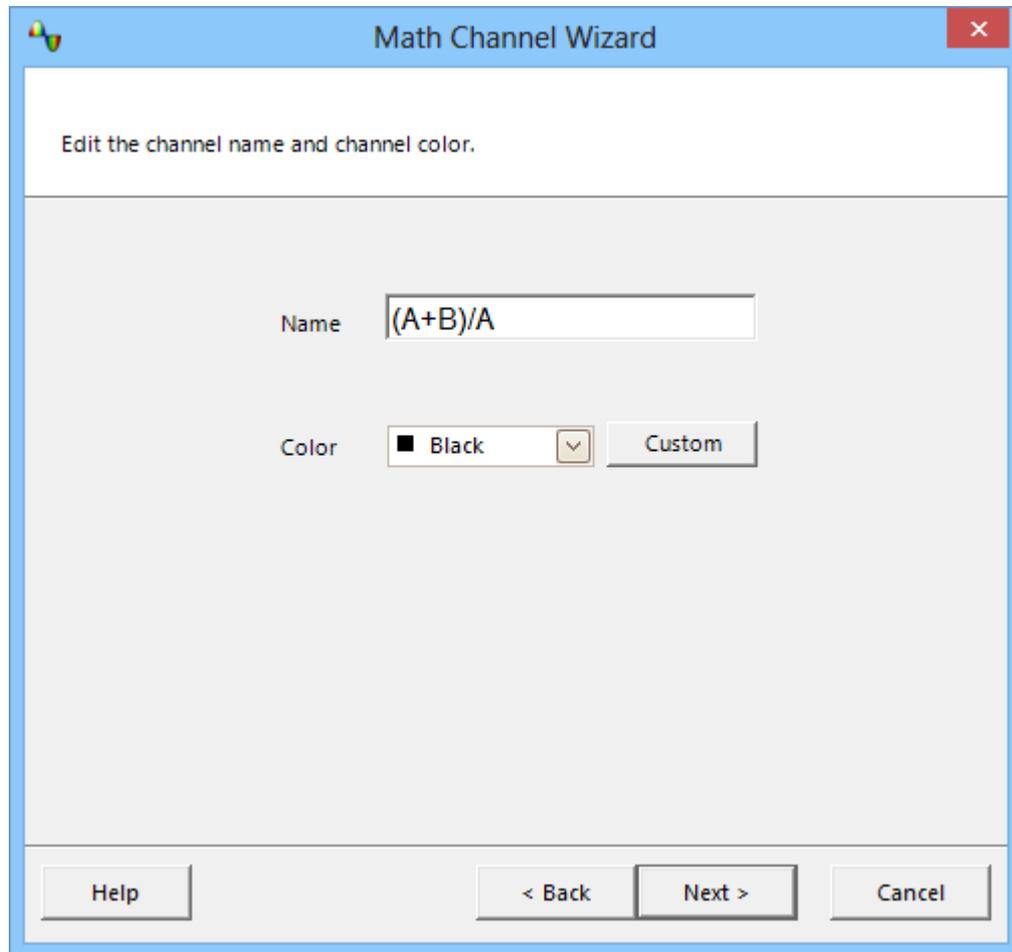
Signum function. The **sign()** operator returns the sign of its input. The result is +1 when the input is positive, -1 when the input is negative, and 0 when the input is 0.

Advance/Delay. To deskew a signal, select the channel and add the deskew time in seconds, in square brackets. For example, to advance Channel A by 1 millisecond, enter **A[0.001]**, and to delay Channel A by 1 millisecond, enter **A[-0.001]**. Using time offsets to adjust one channel relative to another can compensate for differences in cable lengths or skew in trace lengths.

6.5.2.1.3 Math Channel Wizard Name dialog

Location: [Math Channel Wizard](#)

Purpose: allows you to enter or edit the name and color of a [math channel](#)



PicoScope initially sets the name to the text of the equation, but you can edit it to anything you like. The name will appear in the channel list in the [Math Channels dialog](#). You can set the color of the trace to one of the standard colors in the drop-down-list, or click **Custom** to choose any possible color allowed by Windows.

6.5.2.1.4 Math Channel Wizard Units and Range dialog

Location: [Math Channel Wizard](#)

Purpose: allows you to specify the measurement units and the range of values to display for a [math channel](#)

Math Channel Wizard

Select the range and units for your Maths Channel.

Units

Long Name Short Name

Range

Override automatic range selection.

Min Max

Help < Back Next > Cancel

Units, Long Name: This is for your reference only.

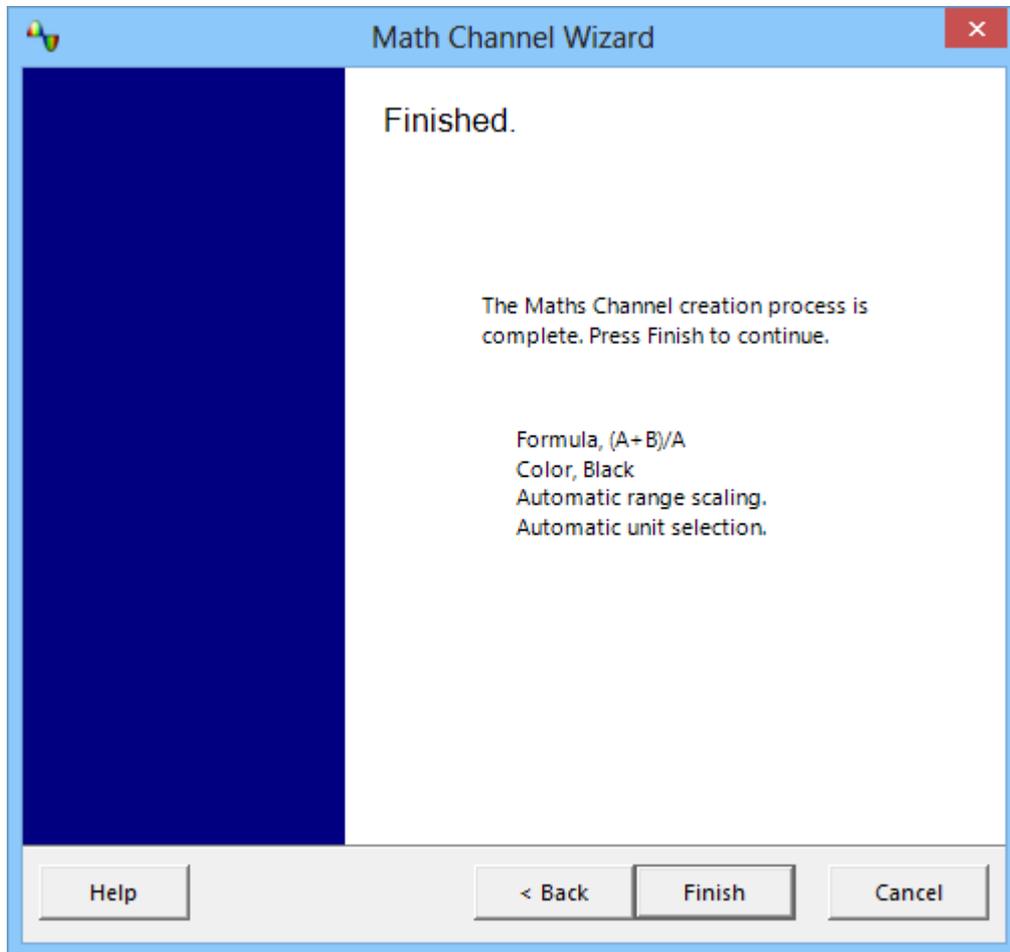
Units, Short Name: This will be displayed on the measurement axis in [scope](#) and [spectrum](#) views, in the [ruler legend](#) and in the [measurements table](#).

Range: If you leave the check box empty, PicoScope will choose the most appropriate range for the measurement axis. If you prefer to set your own values for the minimum and maximum extremes of the measurement axis, check the **Override automatic range selection** box and enter them in the **Min** and **Max** boxes.

6.5.2.1.5 Math Channel Wizard Finished dialog

Location: [Math Channel Wizard](#)

Purpose: shows you the settings for the [math channel](#) that you have just created or edited



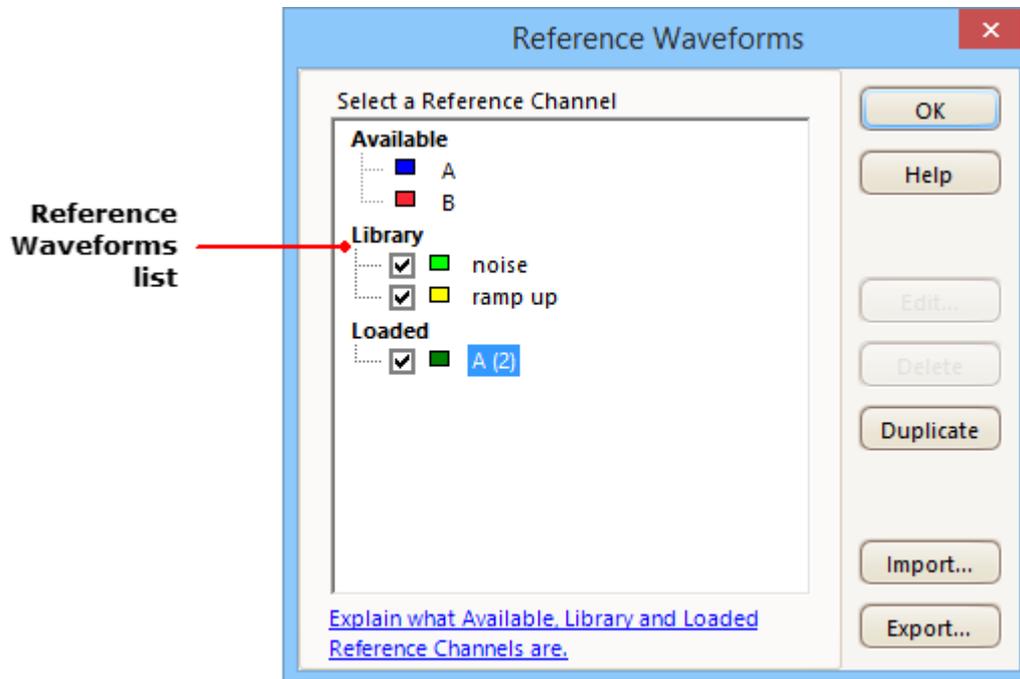
Back. Click this button to return to previous dialogs in the [Math Channel Wizard](#) if you wish to change any of the settings.

Finish. Click this button to accept the settings shown and return to the [Math Channels dialog](#). If you want the new or edited channel to appear on the scope or spectrum display, remember to tick the appropriate check box in the channel list. You can change them later by clicking the **Math Channels button** in the [Channel Setup toolbar](#).

6.5.3 Reference Waveforms dialog

Location: [Tools](#) > [Reference Waveforms](#)

Purpose: enables you to create, [edit](#) and control [reference waveforms](#), which are stored copies of input channels



Reference Waveforms list

The main area of the **Reference Waveforms dialog** is the **Reference Waveforms list**, which shows all the available input channels and the library and loaded [reference waveforms](#). To choose whether or not a waveform appears in the main [PicoScope window](#), click the appropriate check box and then **OK**. You can have up to 8 channels in any view, including input channels, math channels and reference waveforms. If you try to enable a 9th channel, PicoScope will open another view.

Available: these input channels are suitable as sources for reference waveforms

Library: these are the reference waveforms that you have defined using the **Duplicate** button or loaded with the **Import** button

Loaded: these are the reference waveforms present in any PicoScope settings or data files that you have loaded

Edit

Opens the [Edit Reference Waveform dialog](#) to allow you to edit the selected reference waveform. You must first select a waveform in the **Library** section of the **Reference Waveforms list**. If the waveform you want to edit is in the **Loaded** section, first copy it to the **Library** section by clicking **Duplicate**, then select it and click **Edit**.

Delete

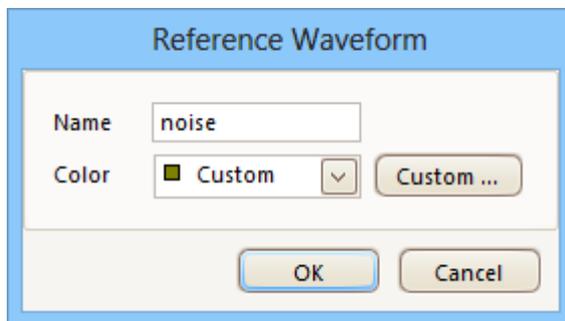
Permanently deletes the selected reference waveform. Only reference waveforms in the **Library** section can be deleted.

- Duplicate** Creates a copy of the selected input channel or reference waveform. The copy is placed in the **Library** section, from where you can edit it by clicking **Edit**. A quicker way to do the same thing is to right-click on the view, select **Reference Waveforms** and then click the channel that you wish to copy.
- Import** Opens a `.psreference` reference waveform file and places the waveforms it contains in the **Library** section.
- Export** Saves all reference waveforms from the **Library** section to a new `.psreference` or MATLAB 4 `.mat` file.

6.5.3.1 Edit Reference Waveform dialog

Location: [Reference Waveforms dialog](#) > **Edit**

Purpose: allows you to edit the name and color of a [reference waveform](#)

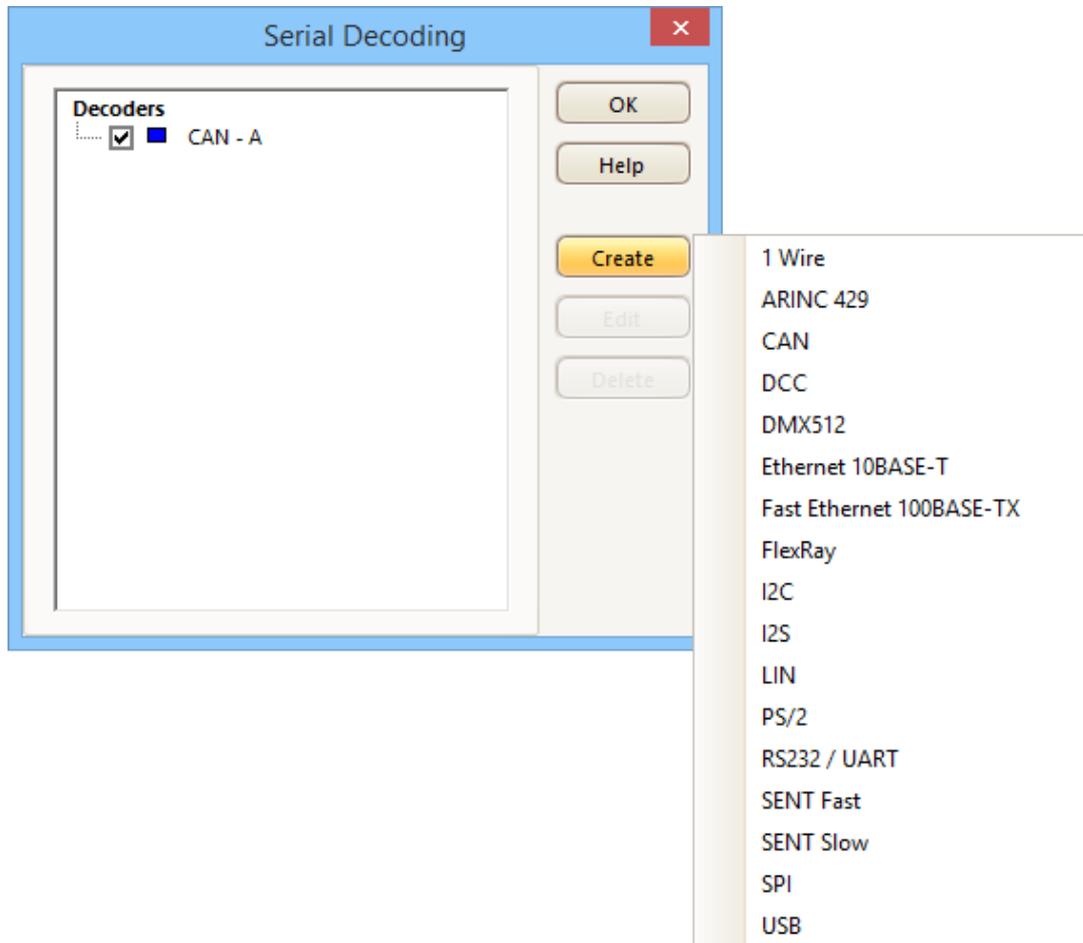


- Name.** PicoScope initially names the waveform after the input channel used as its source, but you can edit it to anything you like. Here we have named it *sine*. The name will appear in the waveform list in the [Reference Waveforms dialog](#).
- Color:** You can set the color of the trace to one of the standard colors in the drop-down-list, or click **Custom** to choose any color allowed by Windows.

6.5.4 Serial Decoding dialog

Location: [Tools](#) > **Serial Decoding**

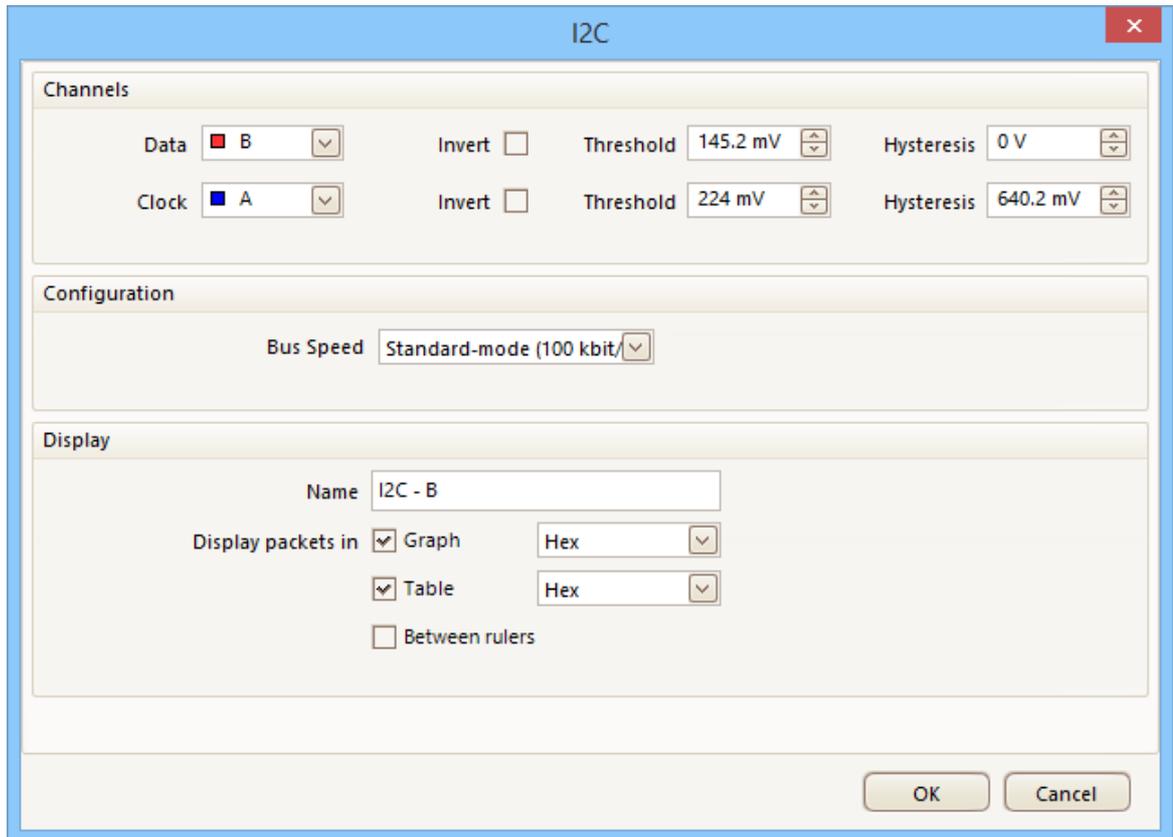
Purpose: lets you set up which channels and protocols to use for [serial decoding](#)

**Adding a new decoder**

1. Click [Tools](#) > **Serial Decoding** from the Menu bar.
2. Click **Create** from the Serial Decoding dialog.
3. Select the protocol you would like to use to decode your data, from the list shown above.
4. Select which channel to decode, how you would like to view the data, and adjust other settings in the **Protocol Settings** dialog. Each protocol has a different set of settings to choose from. See [Protocol Settings](#) for more information.
5. If the channel you want to use is not listed, enable it first by using the [Channels toolbar](#). For duplex protocols like UART, you must decode each direction (Tx and Rx in this case) using a separate channel.
6. Click **OK**. Your selected protocols will appear in the Serial Decoding list, where they can later be edited or deleted.

6.5.4.1 Protocol Settings

For each protocol, different settings options are available



Channels

Controls

Channel

The channel attached to the specified signal. Configure this to match the connection of the inputs to the scope device.

Invert

Reverse the polarity of the signal.

Threshold

The voltage that defines the transition in either direction between high and low logic states.

Hysteresis

The voltage difference between the two thresholds used to reject noise.

Signals

Data

Data channel for various protocols.

Note: bidirectional SPI can be decoded as two separate buses, one with Data In (DI) as the I/O signal and the other with Data Out (DO) as the I/O signal. The Clk (and Chip Sel if present) can be shared by both buses.

Clock

Clock for various protocols.

Word Select

For the I²S protocol. Indicates which stereo channel the data belongs to.

Chip Select

For the SPI protocol. If there is no Chip Select signal, clear the check box.

DataPlus and **DataMinus**

For the USB protocol.

Configuration**Baud Rate**

The symbol frequency in baud. Choose one of the standard values from the drop-down list, type in an arbitrary value, or click  to detect the baud rate automatically.

Bit Order

Whether the least-significant or the most-significant bit occurs first.

For protocols with application-specific bit order, such as UART, you must set this to match to the data format.

Bus Speed

The maximum data rate of the I²C serial bus.

Chip Select State

The polarity of the Chip Select (CS) signal.

Clock Edge

Whether to sample the clock (SCK) on the falling or rising edge.

Data Bits

The number of bits in the data payload of each packet.

Data Encoding

Encoding of the ARINC 429 data field: Binary Coded Decimal (BCD), Binary Number Representation (BNR), or Discrete data representation.

Display 4B/5B Code Groups

Show the 4B/5B Block Coding scheme output (rather than Ethernet packets) for the Fast Ethernet protocol.

First Slot

The first DMX channel to display.

High or Low

The signal polarity of the CAN protocol.

High Threshold

The voltage level of the high signal state.

Last Slot

The last DMX channel to display.

Low Threshold

The voltage level of the low signal state.

Parity

The type of error-correction bit added to each packet.

Sample time from falling edge

The time since the last falling edge at which to sample the 1-Wire bus.

Sensor Type

The type of SENT sensor attached. Defines the contents of the packet data nibbles.

Stop Bits

The number of fixed bits at the end of each packet.

Tick Time

SENT transmitter-specific nominal clock period (tick).

Display**Name**

Create a name for each protocol setup, which is displayed in the scope view.

Display packets

Graph: shows the data in logic-analyzer style, on the same time axis as the analog waveform.

- Hover the mouse pointer over any decoded packet to display its contents.
- Click-and-drag the decoded data up or down the scope view.
- If the **Table** display is visible, double-click on any packet to highlight it in the table.

Table: displays the decoded data in a table in the [serial data window](#), with advanced search and filtering functions.

Between rulers: Only decodes the section of waveform shown between the time rulers.

You can also choose a format for the decoded data: **Hex**, **Binary**, **Decimal**, or **ASCII**.

If the serial decoding parameters entered are incorrect, PicoScope will display an error message.

6.5.4.2 Serial Decoding data window

Select **Table** in the [Protocol Settings](#) dialog to view the decoded serial data in an alphanumeric format below your waveform, as shown below. This format allows advanced filtering and searching of the data.

When **Graph** is also selected in the [Protocol Settings](#) dialog, the data will appear in graphical form on the scope view too. You can click on one of the data packets in the scope view to go to the corresponding row in the table, or you can double-click a row in the table to zoom to the corresponding frame in the scope view.

Control bar

Fields

| Packet | Start Time | End Time | Address Bits | Address | Address + R/W | R/W |
|--------|------------|----------|--------------|---------|---------------|-------|
| 13 | 7.342 ms | 7.938 ms | 7 | 29 | 52 | Write |
| 14 | 7.938 ms | 8.514 ms | 7 | 29 | 53 | Read |
| 15 | 8.556 ms | 9.152 ms | 7 | 2A | 54 | Write |
| 16 | 9.152 ms | 9.728 ms | 7 | 2A | 55 | Read |
| 17 | 9.839 ms | 10.43 ms | 7 | 2A | 54 | Write |
| 18 | 10.43 ms | 11.01 ms | 7 | 2A | 55 | Read |
| 19 | 11.05 ms | 11.65 ms | 7 | 2A | 54 | Write |
| 20 | 11.65 ms | 12.22 ms | 7 | 2A | 55 | Read |

Selection tabs

Fields

The fields shown will vary depending on which protocol has been selected. For more information on the data displayed, see the individual [Serial protocols](#) sections.

Click and drag the edges of each column to change the width, and make viewing the field data clearer.

Selection tabs

A separate table will be displayed for each serial decoding setup created, with different channels and protocols. Use the tabs to switch between them.

Control bar

Export Save the decoded data in a link file, as an OpenDocument Spreadsheet (ODS) or Comma-Separated Values (CSV) text file.

Link **Open:** Open a previously created link file, which translates numerical decoded data to text.

Create Template: Create a link file with the correct field headers, but without any populated data.

Clear: Clear the loaded link file.

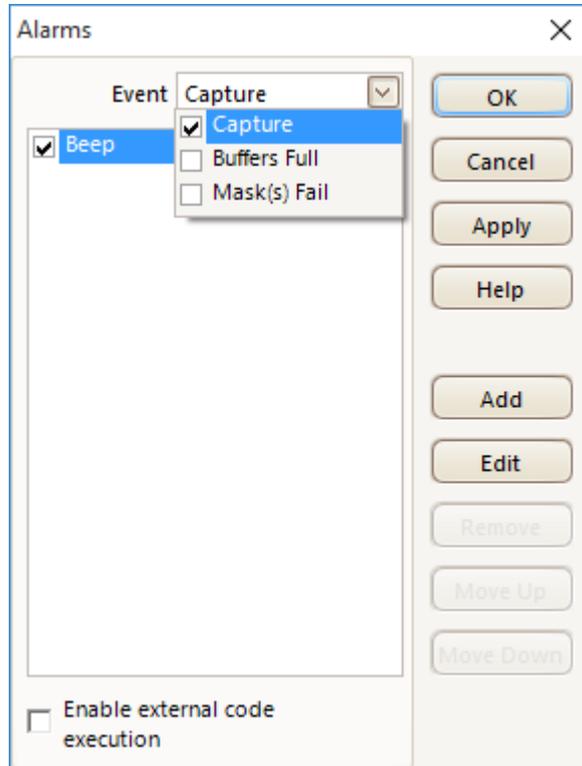
Recent Files: Files which have recently been saved or opened.

| | |
|----------------|--|
| View | <p>Fields: Choose which data fields are displayed.</p> <p>Grid Font Size: Change the size of the characters in the table.</p> |
| Statistics | View or hide the statistics column, which lists measurements such as packet start and end times and signal voltages. |
| Display format | Choose the format your data is displayed in: Hex, Binary, Decimal, or ASCII. |
| Buffer | Choose how much data is displayed: from the Current Buffer or All Buffers. |
| Filter | Click to show the filter bar, which allows you to enter a data search term above each column in the table. Only the packets which match the data search term you have entered will be displayed. For example, enter 6C7 in the filter box at the top of the ID column and PicoScope will only display the packets with an ID of 6C7. |
| Search | Select the field you wish to search using the drop-down list, and then enter a search term in the search box. Use the arrows to search through matching data packets. |

6.5.5 Alarms dialog

Location: [Tools](#) > **Alarms**

Purpose: gives access to the alarms feature, which specifies actions to be taken on various events



Event: Select the event(s) that will trigger the alarm:

Capture: when a waveform is captured. If [triggering](#) is enabled, this option corresponds to a trigger event. You can therefore use this function to save a file on each trigger event.

Buffers Full: when the number of waveforms in the [waveform buffer](#) reaches the [maximum waveform count](#).

Mask(s) Fail: when any channel fails a [mask test](#).

NOTE: For an event to be active, its check box must be selected.

(Actions list) Add actions to this list by clicking **Add**. Whenever the event selected above occurs, PicoScope will execute all the actions in the list from top to bottom. For an action to execute, its check box must be selected.

Apply Set up the scope according to the settings in this dialog.

Add Add an event to the **Actions** list. Possible events are:

Beep: activate the computer's built-in sounder. 64-bit PCs redirect this sound to the headphone output.

Play Sound: specify the name of a `.wav` sound file to play.

Stop Capture: equivalent to pressing the red **Stop** button.

Restart Capture: equivalent to pressing the green **Start** button. Use only if the **Stop Capture** action was used earlier in the list.

Run Executable: run the specified EXE, COM or BAT program file. You can type the `%file%` variable after the program name to pass the name of the last file saved as an argument to the program. PicoScope will stop capturing while the program runs, and resume after the program terminates.

Save Current Buffer: save the current waveform from the buffer as a `.psdata`, `.pssettings`, `.csv` or `.mat` file. You can use the `%buffer%` variable to insert the buffer index number into the filename, or the `%time%` variable to insert the time of capture.

Save All Buffers: save the entire waveform buffer as a `.psdata`, `.pssettings`, `.csv` or `.mat` file.

Trigger Signal Generator: if the scope device has a [triggerable signal generator](#), start generating a signal. The signal generator's Trigger Source option must be set to **Manual** for this operation to be effective.

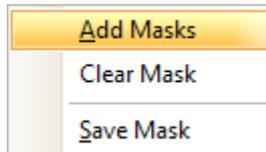
Enable external code execution

As a security feature, to enable **Run Executable** you must also select the **Enable external code execution** option.

6.5.6 Masks menu

Location: [Tools](#) > **Masks**

Purpose: gives control over [Mask Limit Testing](#)



Add Masks Add a mask to the display using the [Mask Library dialog](#).

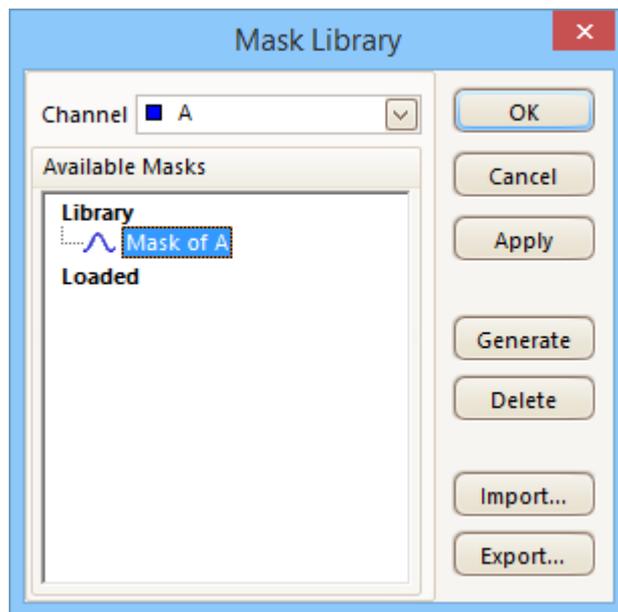
Clear Mask Remove the mask from the display.

Save Mask Save the displayed mask to disk as a `.mask` file.

6.5.6.1 Mask Library dialog

Location: [Tools](#) > **Masks**

Purpose: allows you to create, export and import masks for [Mask Limit Testing](#)



Channel Select the channel to which you want to apply the mask.

Available masks The **Library** section shows all the masks that you have saved in the past and not deleted. The **Loaded** section shows all the masks currently in use.

Generate Create a new mask based on the last waveform captured from the selected channel. Opens the [Generate Mask dialog](#).

Import Load a mask that has previously been saved as a `.mask` file.

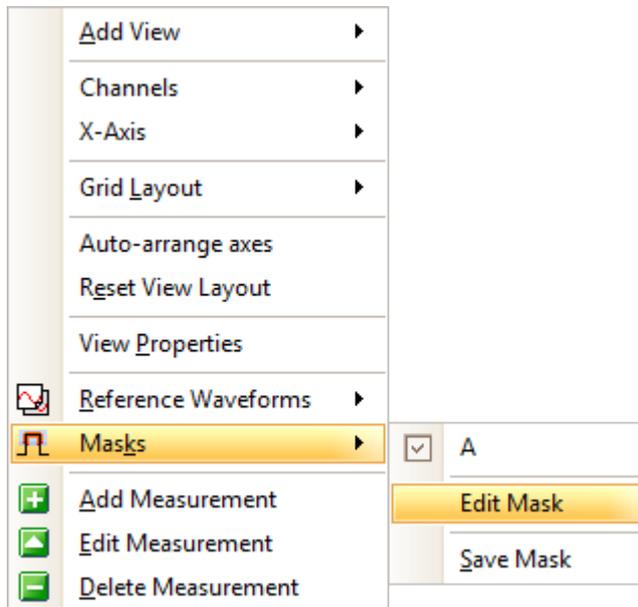
Export Save a mask as a `.mask` file for future import.

Apply Use the selected mask on the selected channel but remain in the **Mask Library dialog**.

OK Use the selected mask on the selected channel and return to [scope view](#).

6.5.6.2 Editing a mask

To edit a mask in **Mask Limit Testing** mode, right-click on the [scope view](#) and select **Masks > Edit Mask:**

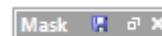


A mask is made up of one or more shapes called **polygons**. Click on the polygon you wish to edit. PicoScope will then draw edit handles on the selected mask polygon and display the mask edit box. If you drag any of the handles to edit the polygon, the statistical results will be updated immediately.

The mask edit box looks like this:

| X | Y |
|---------------|----------|
| 500 μ s | 2 V |
| 0 s | 2 V |
| 0 s | 1.185 V |
| 44.5 μ s | 1.183 V |
| 45 μ s | 1.167 V |
| 45.75 μ s | 1.167 V |
| 47.75 μ s | 1.131 V |
| 48.25 μ s | 1.115 V |
| 49.25 μ s | 1.114 V |
| 50.25 μ s | 1.078 V |
| 51.75 μ s | 1.056 V |
| 52.5 μ s | 1.028 V |
| 53.25 μ s | 1.018 V |
| 58.25 μ s | 849.5 mV |

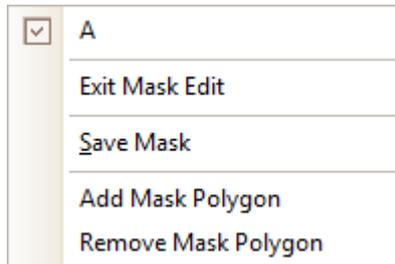
Normal view



Minimized view

If the edit box is not immediately visible, it may have been minimized; in which case, click the restore button: . If you edit the coordinates of a vertex, the statistical results will be updated immediately. You can also export the mask to a `.mask` file with the export button: . Use the **+** and **-** buttons to add or remove vertices. The minimize button has its usual function. To leave mask editing mode, close the mask edit box with the close (**X**) button.

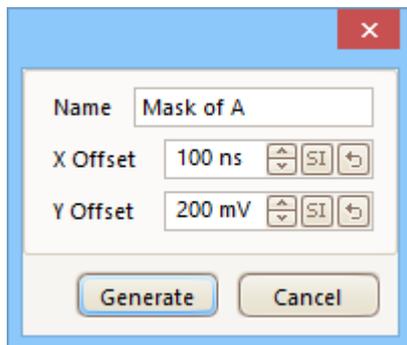
To add or remove an entire polygon, right-click the scope view and select either the **Add Mask Polygon** or the **Remove Mask Polygon** command:



6.5.6.3 Generate Mask dialog

Location: [Mask Library dialog](#) > **Generate**

Purpose: allows you to set parameters for the automatically generated mask. PicoScope will then create a new mask based on the last captured waveform.



Name: PicoScope automatically chooses a name for the new mask. You can edit the name in this box.

X Offset: The horizontal distance between the waveform and the mask.

 /  This button toggles the offset value between absolute units (SI) and relative units (% of full scale).

 This button resets the offset value to its default.

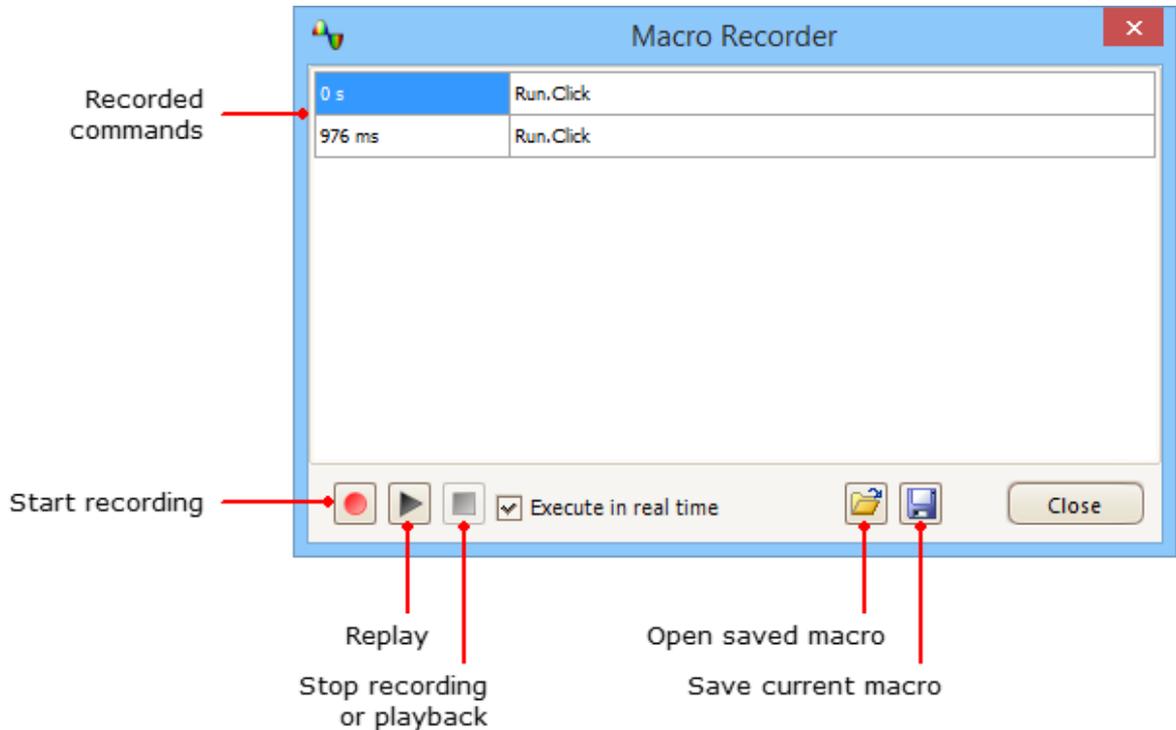
Y Offset: The vertical distance between the waveform and the mask.

6.5.7 Macro Recorder

Location: [Tools](#) > **Macro Recorder**

Purpose: records a sequence of commands to be replayed later

The **Macro Recorder** helps when you wish to execute a series of commands repeatedly. It saves all commands to a `.psmacro` file, which can be modified using an XML editor.



Execute in real time Play back the macro at the same speed as when recorded. Without this option, playback will be as fast as possible.

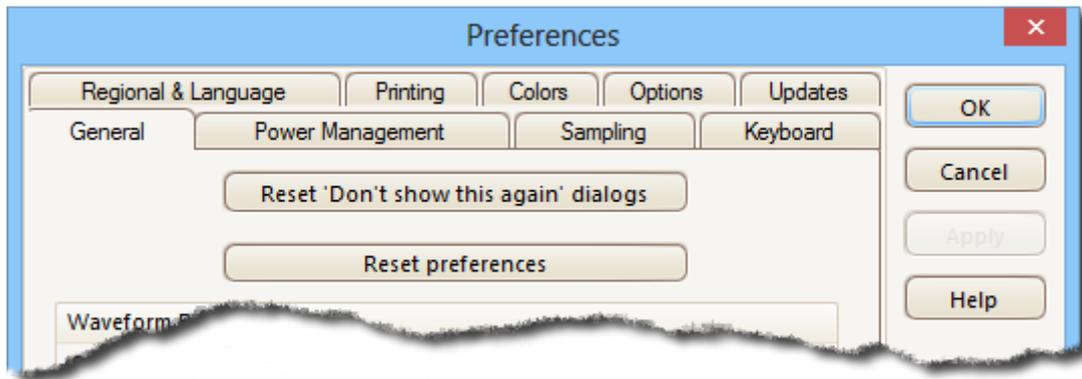
Note 1: `.psmacro` files can also be played back from the [PicoScope command line](#).

Note 2: You can obtain a list of macro commands from the PicoScope command line. See [Command-line syntax](#).

6.5.8 Preferences dialog

Location: [Tools](#) > **Preferences**

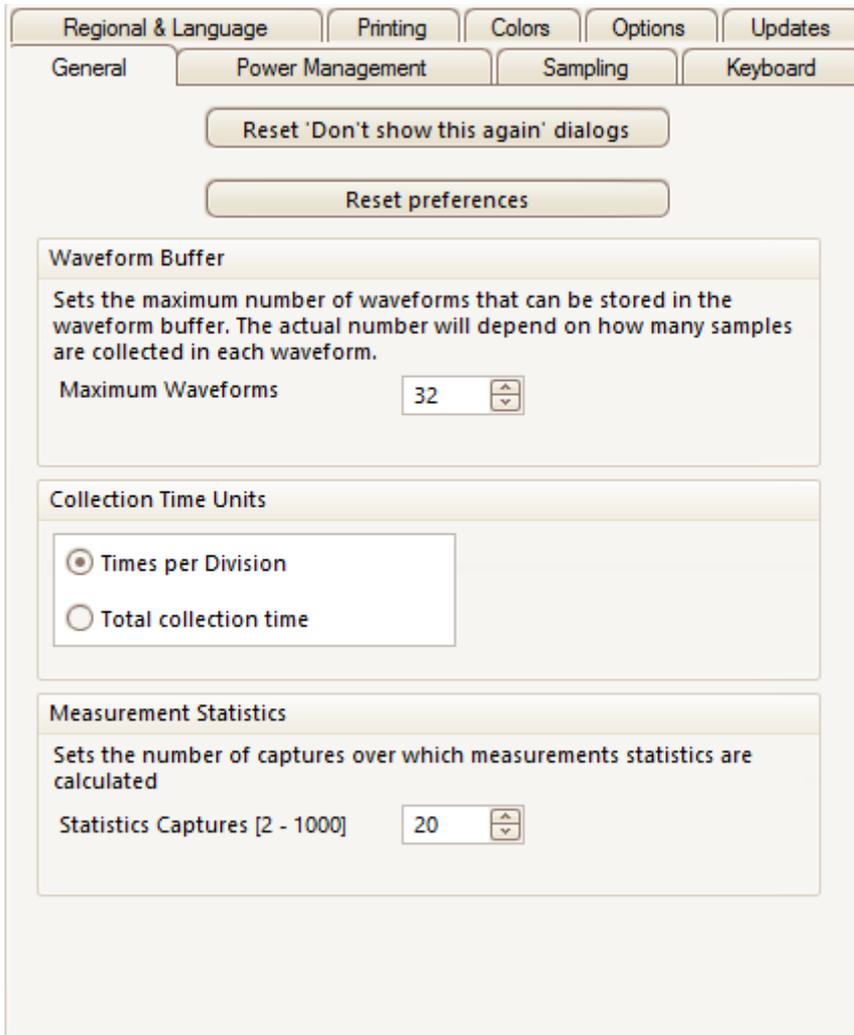
Purpose: Allows you to set options for the PicoScope software. Click one of the tabs in the picture below to learn more.



6.5.8.1 General page

Location: [Tools](#) > [Preferences](#) > **General**

Purpose: contains general controls for PicoScope

**Reset "Don't show this again" dialogs**

Restore any missing dialogs that you asked PicoScope not to show again.

Reset preferences

Set all preferences back to their default values.

Waveform Buffer

Maximum Waveforms: This is the maximum number of waveforms that PicoScope will store in the [waveform buffer](#). You can select a number from 1 to the maximum allowed by the oscilloscope that is connected: see the scope specifications for details). The actual number of waveforms stored depends on the available memory and the number of samples in each waveform.

Collection Time Units

Change the mode of the **Timebase** control in the [Capture Setup toolbar](#).

Times per division: the **Timebase** control displays time units per division - for example, 5 ns /div. Most laboratory oscilloscopes display timebase settings in this way.

Total collection time: the **Timebase** control displays time units for the entire width of the scope view - for example, 50 ns.

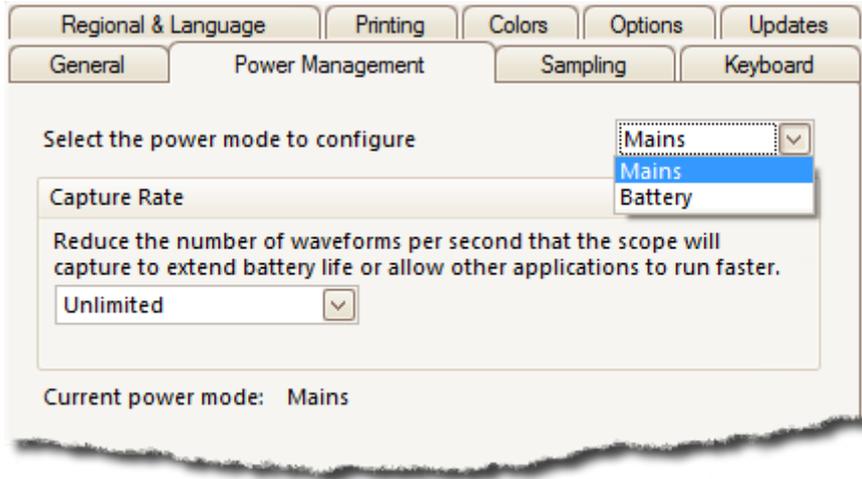
Measurement Statistics

Capture Size - the number of successive captures that PicoScope uses to calculate the statistics in the [Measurements Table](#). A larger number produces more accurate statistics but causes them to be updated less frequently.

6.5.8.2 Power management page

Location: [Tools](#) > [Preferences](#) > **Power Management**

Purpose: controls features of the oscilloscope that affect its power consumption



Capture Rate

This control limits the speed at which PicoScope captures data from the scope device. The other PicoScope settings, the type of [scope device](#) and the speed of the computer will all affect whether this limit can actually be reached. PicoScope automatically selects the appropriate limit according to whether your computer is running on batteries or on mains (line) power.

The settings are in captures per second. By default, the capture rate is set to *Unlimited* when your computer is running on **Mains** (line) power, for maximum performance. If other applications run too slowly on your PC whilst PicoScope is capturing, then reduce the capture rate limit. When your computer is running on **Battery** power, PicoScope imposes a performance limit to save the battery. You can increase this limit manually, but this will cause the battery power to drain very quickly.

6.5.8.3 Sampling page

Location: [Tools](#) > [Preferences](#) > **Sampling**

Purpose: controls the sampling behavior of the oscilloscope

Regional & Language Printing Colors Options Updates

General Power Management **Sampling** Keyboard

Slow Sampling Transition

Sets the collection time at which PicoScope 6 will try to go into slow sampling mode. This is the mode where PicoScope 6 will update the screen before the total collection time has expired.

Collection Time ▾

Depending on device specifications, PicoScope 6 may not be able to go into slow sampling mode at the required collection time, in which case the nearest collection time possible will be used.
The slow sampling transition for the current Device: 200 ms/div

Slow Sampling Display

Display previous Waveform Buffer

Sin(x)/x Interpolation

On Off

Specify sample threshold at which interpolation activates when operating at the fastest sample rate; applicable only in scope mode.

▴ ▾

Slow Sampling Transition

In normal (fast) sampling mode, PicoScope collects enough data to fill the screen and then redraws the whole view at once. This method is suitable for fast timebases, when the screen is redrawn many times each second, but with slow timebases it can cause an unacceptable delay before the data appears on the screen. To avoid this delay, PicoScope automatically switches to slow sampling mode, in which the scope trace advances gradually across the screen as the scope captures data.

The **Collection Time** control lets you select the timebase at which PicoScope switches to slow sampling mode.

Slow Sampling Display

When this box is checked, PicoScope displays the previous waveform in the buffer while gradually redrawing the new waveform over the top of it. Thus, at any time, the left-hand side of the view shows the beginning of the new waveform, while the right-hand side shows the end of the previous waveform. A vertical bar separates the two waveforms. This operation uses the Fast Streaming Mode capability of the PicoScope oscilloscope hardware.

Sin(x)/x Interpolation

When the number of pixels across the scope view is greater than the number of samples in the waveform buffer, PicoScope interpolates - that is, it fills the space between the samples with estimated data. It can either draw straight lines between the samples (linear interpolation) or connect them with smooth curves (sin(x)/x interpolation). Linear interpolation makes it easier to see where the samples are, which is useful for high-accuracy measurements, but results in a jagged waveform. Sin(x)/x interpolation gives a smoother waveform but disguises the true locations of the samples, so should be used with care when the number of samples on the screen is low.

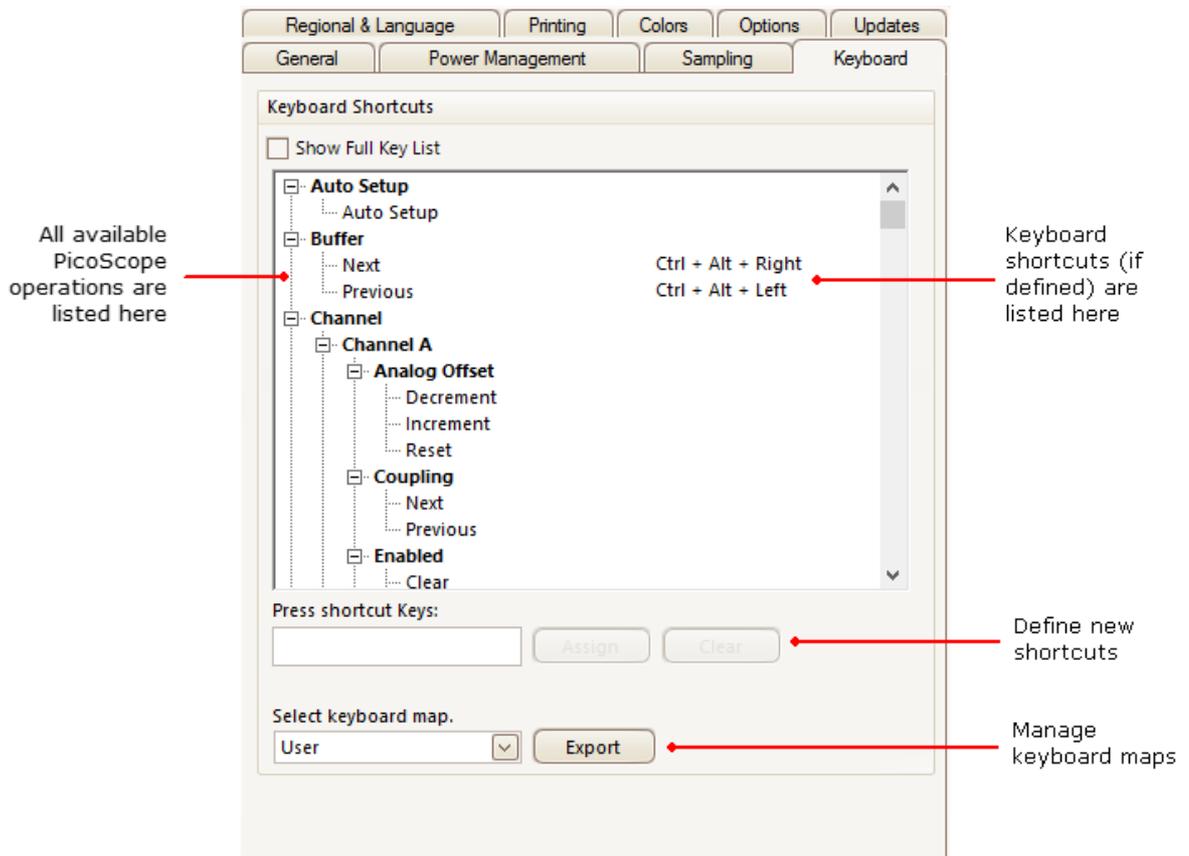
You can adjust the number of samples below which sin(x)/x interpolation is switched on. Sin(x)/x interpolation is used only on the scope's fastest timebase.

6.5.8.4 Keyboard page

Location: [Tools](#) > [Preferences](#) > **Keyboard**

Purpose: displays, and allows you to edit, keyboard shortcuts

A keyboard shortcut is a combination of keys that can be pressed on the keyboard to activate a PicoScope operation.



Keyboard Shortcuts This is a list of PicoScope operations and their associated keyboard shortcuts (if defined). The extent of the list depends on the [Show Full Key List](#) option (see below).

To edit or add a keyboard shortcut:

- Scroll the list of PicoScope commands until the required operation is visible.
- Select the required operation.
- Select the *Press shortcut keys:* box.
- Press the required key combination on the keyboard.
- Click **Assign**.

Show Full Key List Check this box to show all available operations. By default, only the most common operations are listed, plus any other operations that have a keyboard shortcut assigned to them.

Keyboard Maps

A set of keyboard shortcuts is called a **map**. You can define multiple maps for different applications.

Default: This map cannot be edited. Use it to return to the factory-defined basic shortcuts.

Advanced: This is another factory-defined map that cannot be edited. It contains a more comprehensive set of shortcuts.

User: This is the map that you most recently created or imported. It is preserved between PicoScope sessions.

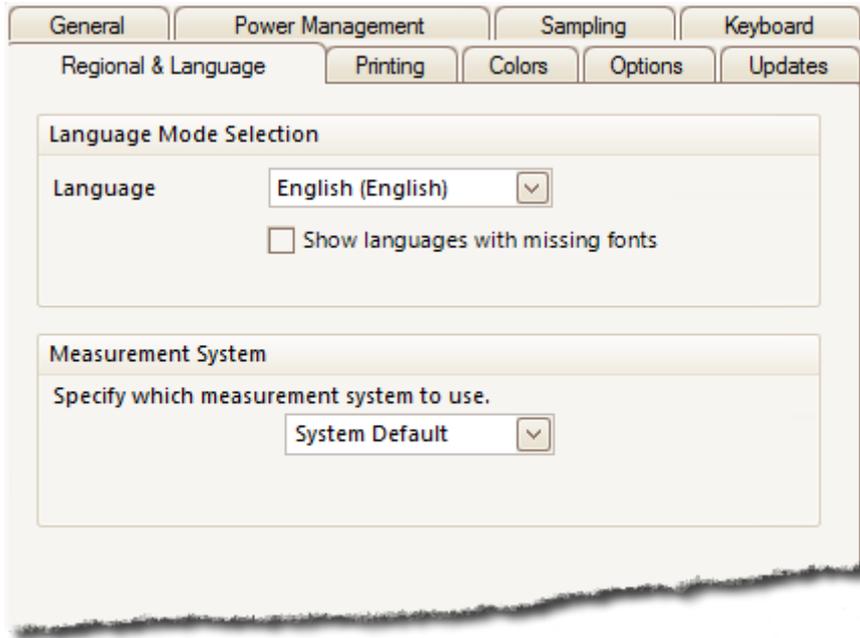
Import: Load a keyboard map from a `.pskeys` file.

Export: Save the current keyboard map to a `.pskeys` file.

6.5.8.5 Regional & Language page

Location: [Tools](#) > [Preferences](#) > **Regional & Language**

Purpose: lets you select the language and other location-dependent settings for PicoScope's user interface

**Language**

Select, from the drop-down box, the language you wish to use for the PicoScope 6 user interface. PicoScope will ask you to restart the program before switching to the new language.

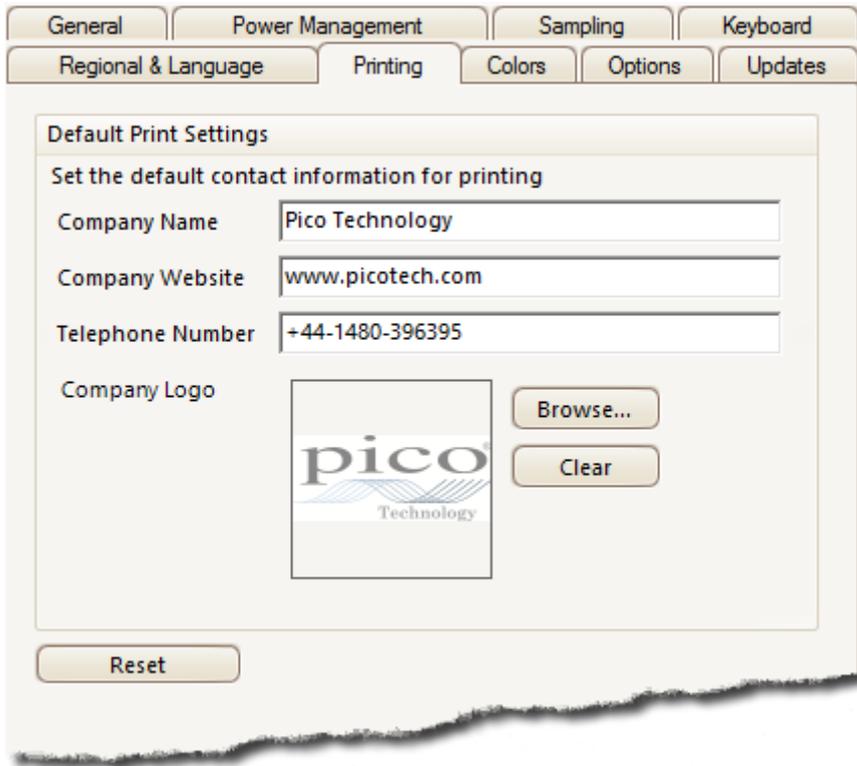
Measurement System

Select metric or U.S. units.

6.5.8.6 Printing page

Location: [Tools](#) > [Preferences](#) > **Printing**

Purpose: lets you enter the details that will appear at the bottom of printed output

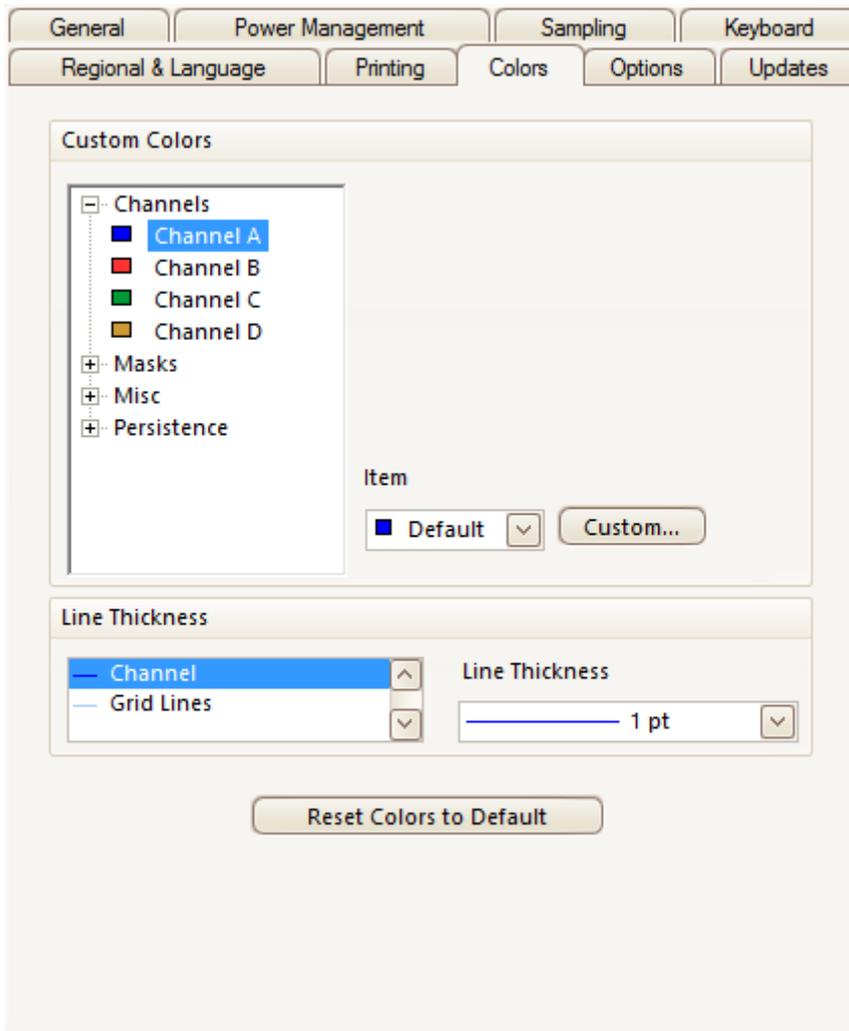
**Default Print Settings**

When you print a view from the [File menu](#), these details will be added to the bottom of the page.

6.5.8.7 Colors page

Location: [Tools](#) > [Preferences](#) > **Colors**

Purpose: lets you set the colors for various parts of the user interface

**Custom Colors**

These controls let you specify the colors for various parts of the PicoScope screen:

- Channels** the trace color for each [scope channel](#)
- Digital Channels** if you have a [mixed-signal oscilloscope \(MSO\)](#), the color of each channel can be set here
- Masks** the mask areas in [Mask Limit Testing](#)
- Misc** miscellaneous items:
- Grid lines** the horizontal and vertical lines on the [graticule](#)
- Background** the area behind the waveforms and graticule. (In [persistence mode](#) this setting can be overridden by the [Persistence Options dialog](#)).
- Live trigger** the [trigger marker](#) for the current trigger position

| | |
|------------------------|---|
| Trigger | secondary trigger marker (appears when the live trigger has moved since the last waveform capture) |
| Horizontal axis | the numbers across the bottom of each view , which usually indicate time measurements |
| Rulers | the horizontal and vertical rulers that you can drag into position to help measure features on the waveform |
| Persistence | the three colors to use for each channel in digital color persistence mode . The top color is used for the most frequently hit pixels, the middle and bottom colors for the less and least frequently hit pixels. |

Line Thickness

These controls let you specify the thickness of the lines drawn on the [scope](#) and [spectrum](#) views:

| | |
|-------------------|---|
| Channel | the waveforms and spectrum traces for all scope channels |
| Grid Lines | the horizontal and vertical lines on the graticule |
| Markers | the horizontal and vertical rulers that you can drag into position to help measure features on the waveform |

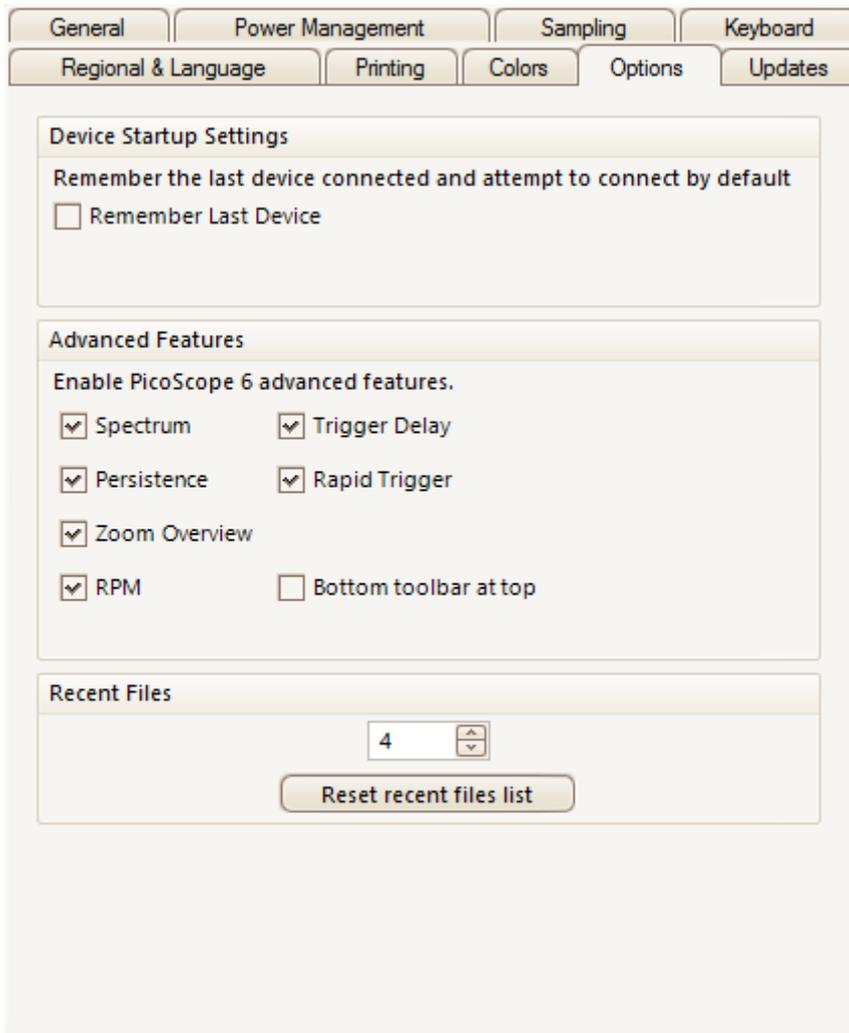
Reset Colors to Default

Resets all of the color and line thickness settings to their default values.

6.5.8.8 Options page

Location: [Tools](#) > [Preferences](#) > **Options**

Purpose: lets you set various options that control the way that PicoScope 6 works



Device Startup Settings

Remember Last Device. This option is used when PicoScope finds more than one scope device. If the box is checked, PicoScope will attempt to use the same device that was used last time. Otherwise it will use the first device available.

Advanced Features

The advanced [capture modes](#) are enabled by default in PicoScope 6 and disabled by default in PicoScope 6 Automotive. Regardless of the version that you have, you can enable or disable these features using the following options:

Spectrum [Spectrum view](#) and [spectrum analyzer](#) features

Persistence Digital Color, Analog Intensity and custom [persistence display](#) modes

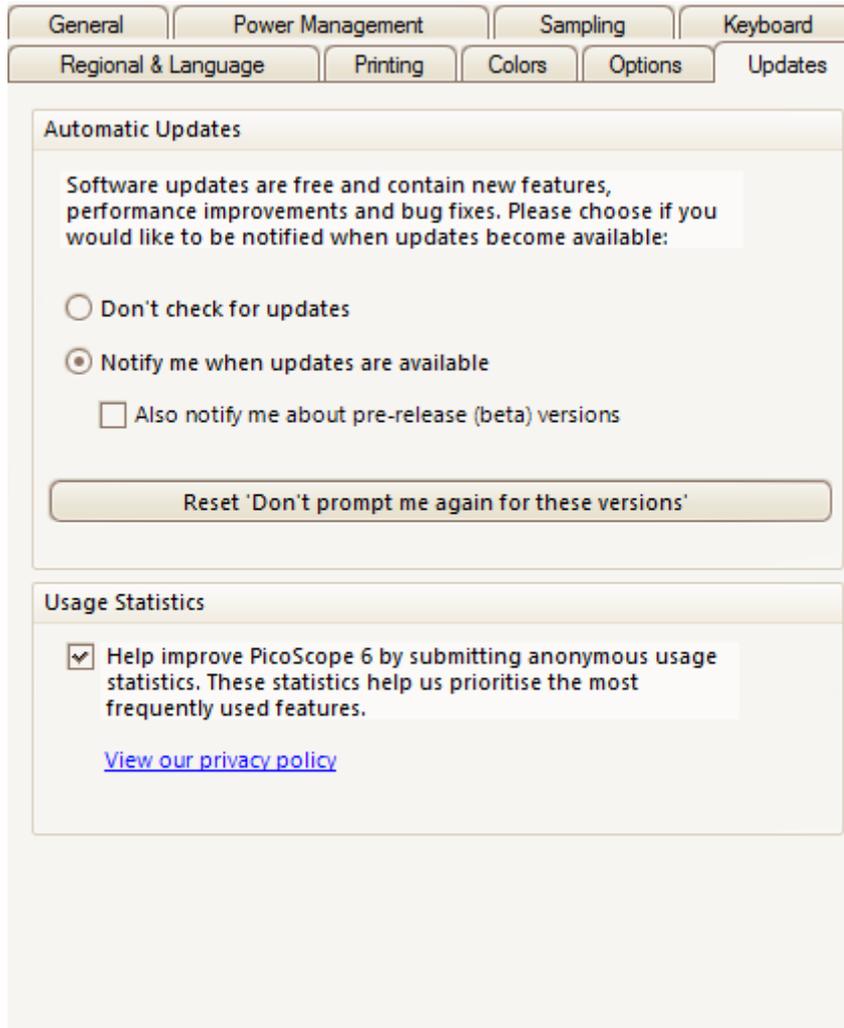
Zoom Overview A window that appears when you [zoom in](#), to help to move around large waveforms with the minimum of mouse clicks

| | |
|------------------------------|---|
| RPM | Revolutions per minute, displayed alongside hertz in the frequency legend |
| Trigger Delay | The time-delay control in the Triggering toolbar . |
| Rapid Trigger | The <i>Rapid</i> entry in the Trigger mode control in the Triggering toolbar . |
| Bottom toolbar at top | The toolbar containing the Start/Stop , Triggering , Measurements and Rulers controls is at the bottom of the PicoScope window by default. This option moves it to the top. |
| Bandwidth Limit | A fixed-frequency single-pole analog filter. |
| Recent Files | The maximum number of files listed in the File > Recent Files menu. Click the button to clear the list. |

6.5.8.9 Updates page

Location: [Tools](#) > [Preferences](#) > **Updates**

Purpose: configures the automatic update checker and related services



Automatic Updates

Don't check for updates If you prefer, you can visit www.picotech.com periodically to check for updates.

Notify me... PicoScope will regularly check for updates to your software. (Requires an internet connection).

Reset... If you checked the *Don't remind me again...* option in the Software Updates dialog, this button enables reminders again.

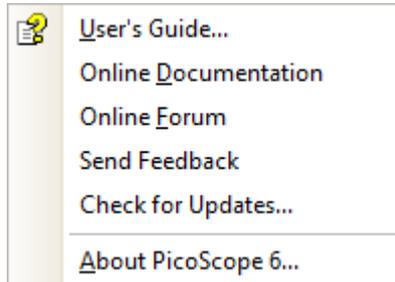
Usage Statistics

To help us to continue improving PicoScope, the program periodically sends back to our developers anonymous information about which features you use the most. This information does not contain your name, email address or any other personal details, but does allow us to identify which country you are in. If you do not wish to send us this information, uncheck this box.

6.6 Help menu

Location: **Help**

Purpose: gives access to the PicoScope 6 User's Guide and related information

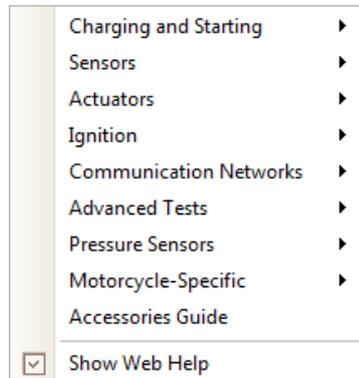


| | |
|-----------------------------|--|
| User's Guide | This is the most complete source of information on PicoScope. |
| Online Documentation | Find manuals and training guides for Pico Technology products. |
| Online Forum | Request technical support and discuss questions with fellow PicoScope users. New software features are sometimes announced here before they make it into the User's Guide . |
| Send Feedback | Send us suggestions for improving PicoScope. |
| Check for Updates | Check online for later versions of PicoScope. If you have both stable and beta versions installed, this checks for updates to both of them. You can configure the settings for automatic update checking under the Tools > Preferences > Updates menu. |
| About PicoScope 6 | Displays useful information such as the model and serial numbers of your scope, and software and driver version numbers. |

6.7 Automotive menu (PicoScope Automotive only)

Location: [Menu bar](#) > **Automotive**

Purpose: gives access to a database of preset tests



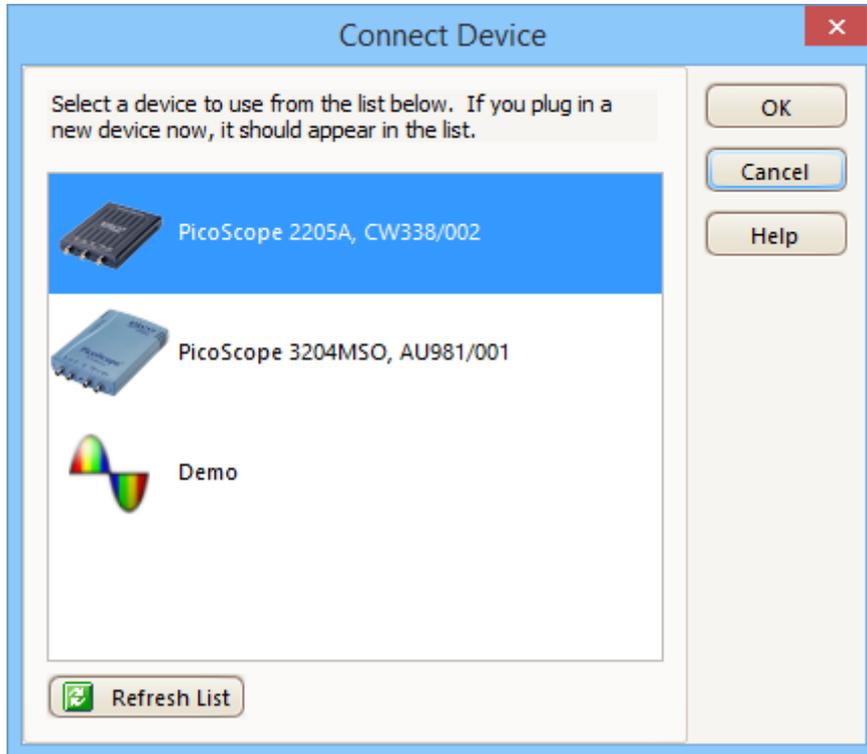
Show Web Help: When this box is checked, selecting a preset test also opens a help file with instructions and technical information. Clear this box to disable.

1. Select a preset test.
2. PicoScope opens the relevant information page that explains how to setup the test for the scope that is connected, run the test and interpret the results (a few tests do not have an information page).
3. PicoScope displays an example waveform.
4. PicoScope configures itself with the necessary settings. In most cases all you need to do is press the space bar to start the test.

6.8 Connect Device dialog

Location: **File > Connect Device**
or plug in a new device

Purpose: when PicoScope finds more than one available [scope device](#), this dialog allows you to select which one to use



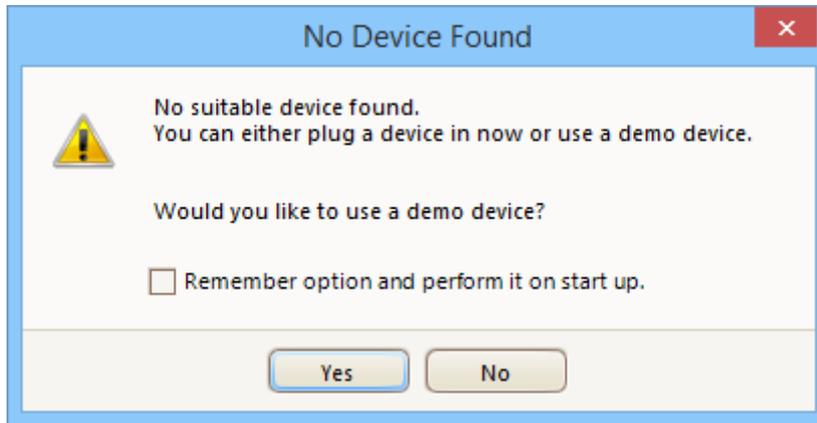
See [How to change to a different device](#) if you wish to switch to a different scope device later.

Procedure

- Wait for a list of devices to appear. This may take a few seconds.
- Select a device and click **OK**.
- PicoScope will open a [scope view](#) for the selected scope device.
- Use the [toolbars](#) to set up the device and the [scope view](#) to display your signals.

6.9 Demo mode

If you start PicoScope without a device connected, the **No Device Found** dialog will automatically appear.



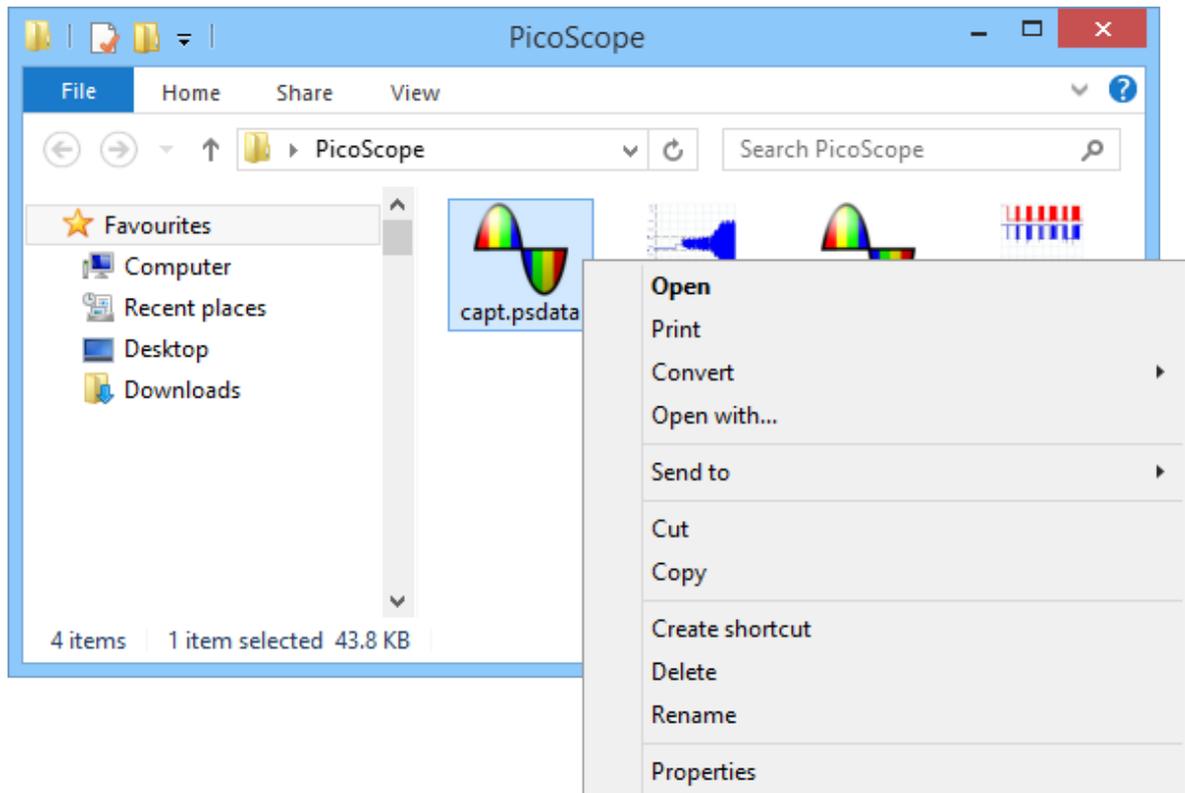
- Select **Yes** to start PicoScope 6 in Demo mode.
- Select **No** to start PicoScope 6 without a signal, allowing access to a limited selection of menu items. Select **File > Connect Device** to change to Demo mode or open another connected device.
- A Demo device can also be selected from the [Connect Device dialog](#).

When in Demo mode, PicoScope adds a [Demo Signal Generator button](#) to the toolbar . Use this button to set up test signals using the Demo device.

6.10 Converting files in Windows Explorer

You can convert PicoScope data files to other formats for use in other applications, or to different forms of data for use with PicoScope.

The easiest way to do this conversion is through the context menu in **Windows Explorer**. The context menu is the menu that pops up when you right-click it with the mouse or activate it with the **menu** button on a Windows keyboard. When you install PicoScope, a **Convert** entry is added to the context menu to enable you to convert PicoScope data files.



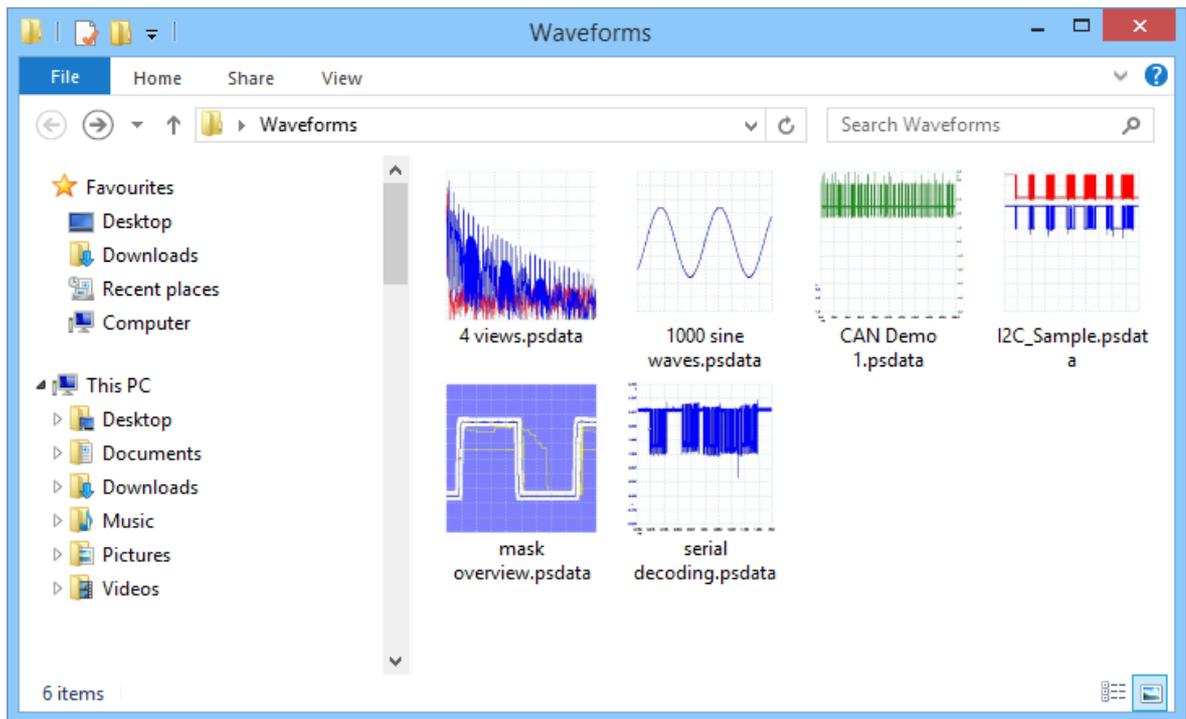
The PicoScope context menu in Windows Explorer

Converting to PicoScope 6.2.4 format

The example above shows four pre-existing PicoScope data files represented by standard PicoScope icons. PicoScope 6.2.4 introduced a new feature that allows PicoScope data files to appear as waveforms instead of icons. To enable this feature for old data files, you need to convert them to the new format using the context menu of Windows Explorer.

- If PicoScope is running, close it.
- In Windows Explorer, right-click a PicoScope data file.
- Select **Convert > All waveforms > .psdata**. A PicoScope icon  will appear in the Windows notification area while the conversion is in progress.
- PicoScope will ask you to confirm that you wish to overwrite the .psdata file with a new version. Click **Yes**.
- Wait for Windows Explorer to update the display.
- Repeat for all .psdata files.

The `.psdata` files should now appear as in this picture:



Converting to other formats

For all of these conversions, you can choose either **All waveforms** or **Current waveform**. A `.psdata` file can contain either a single waveform or the entire contents of the waveform buffer, which can hold a number of waveforms from successive trigger events. If the `.psdata` file contains more than one waveform, then you can choose to convert all of them or just the one that was last viewed in PicoScope.

- Right-click a PicoScope data file.
- To convert all waveforms in the file, select **Convert > All waveforms** or **Convert > Current waveform** and then the file format that you require. A PicoScope icon will appear in the Windows notification area while the conversion is in progress.

Complex operations

For more complex operations, such as converting all files in a directory, you can run PicoScope in a command window (see [Command-line syntax](#)).

7 Toolbars and buttons

A **toolbar** is a collection of buttons and controls with related functions.

7.1 Advanced Options toolbar

The **Advanced Options toolbar** controls **Phase (or Rotation) Rulers, Notes** and (in PicoScope Automotive only) **Channel Labels**.



It contains the following buttons:

| | |
|-----------------------|---|
| Rulers | Opens the Ruler Settings dialog that controls the Phase rulers (or in PicoScope Automotive, Rotation rulers) |
| Notes | Displays the Notes at the bottom of the window |
| Channel Labels | (PicoScope Automotive only) Displays the Channel Labels at the bottom of the window |

This toolbar normally resides at the bottom of the program window, but can be moved to the top using the [Tools > Preferences > Options](#) > **Bottom toolbar at top** control.

7.2 Channels toolbar

The **Channels toolbar** controls the settings for each input [channel](#). The screen shot below shows the toolbar for a two-channel [scope device](#), but different scope devices may have different numbers of channels. Demo device settings are also displayed here. (See also [PicoLog 1216 toolbar](#) that is used for the PicoLog 1000 Series.)

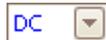


Each channel has its own set of buttons:

Channel Options button. Opens the [Channel Options menu](#) with options for [probes](#), [resolution enhancement](#), [scaling](#) and filtering.



Range control. Sets up the scope device to capture signals over the specified range of values. The list of options depends on the selected [scope device](#) and [probe](#). A red warning symbol -  - appears if the input signal exceeds the selected range. If you select **Auto**, PicoScope will continually adjust the vertical scale so that the height of the waveform fills as much of the view as possible.



Coupling Control. Sets up the input circuitry.

AC coupling: rejects frequencies below about 1 Hz.

DC coupling: accepts all frequencies from DC to the scope's maximum bandwidth.

50Ω DC: low-impedance option (see [Device feature table](#)).

Accelerometer: switches on the current source output for [IEPE](#)-enabled scopes such as the PicoScope 4224 IEPE. The User's Guide for the oscilloscope has details of the [IEPE](#) channel specifications.

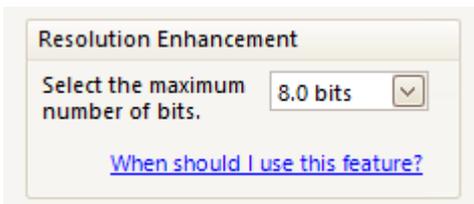
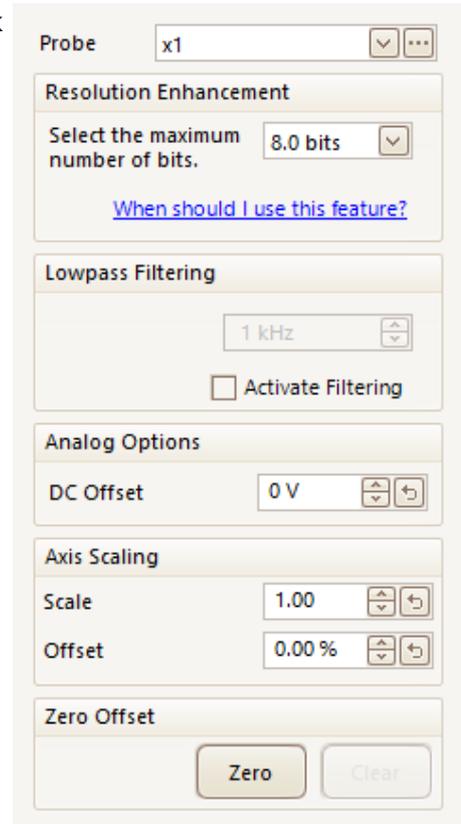
Frequency: enable the built-in frequency counter. Only one channel at a time can be operated in this mode. Available only if your scope device has hardware support for this feature: see [Device feature table](#). Not available in [Demo mode](#).



Digital Inputs button ([MSOs](#) only).

7.2.1 Channel Options menu

The **Channel Options menu** appears when you click the **Channel Options button** (for example: ) on the [Channels toolbar](#).



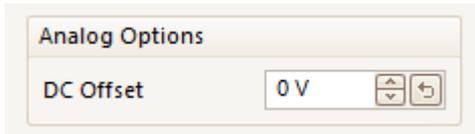
Probe list. Indicates the probe currently in use and allows you to select a different one. Use it to tell PicoScope what type of probe is connected to a channel. By default, the probe is assumed to be x1, which means that a one-volt signal at the input to the probe will appear as one volt on the display.

Expand probe list. Click this to select from a list of probes.

Open Custom Probes dialog. The [Custom Probes dialog](#) allows you to edit your library of custom probes.

Resolution enhancement. Allows you to increase the effective resolution of your scope device using [Resolution enhancement](#). The number in this box is a target value that the software will attempt to use whenever possible.

Lowpass Filtering. An independent digital [lowpass filter](#) for each input channel, with programmable cut-off frequency. This can be useful for removing noise from your signal to make more accurate measurements. For availability, see the [Device feature table](#).

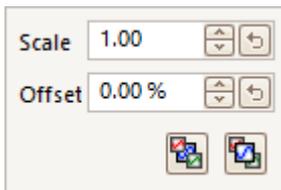


Analog Options. Options that can be applied to the oscilloscope input hardware, if the oscilloscope hardware supports them.

DC Offset: an offset voltage added to the analog input before digitization. For availability, see the [Device feature table](#).

Bandwidth Limit: a fixed-frequency single-pole analog filter. This can be useful for rejecting noise and harmonics that would otherwise cause aliasing. This advanced feature needs to be enabled in [Tools > Preferences > Options](#) before use. For availability, see the [Device feature table](#).

Axis Scaling. These are the [axis scaling controls](#) that allow you to set the scale and offset for each vertical axis individually.



Zero offset. Digitally removes any offset from the input channel. Before starting this operation, remove any input signal from the selected channel and short-circuit the input. Click **Zero** to begin the adjustment. Click **Clear** to restore the input to its uncorrected state.

7.2.1.1 Resolution enhancement

Resolution enhancement is a technique for increasing the effective vertical resolution of the scope at the expense of high-frequency detail. In some scope operating modes PicoScope may reduce the number of samples available to maintain display performance.

For this technique to work, the signal must contain a very small amount of Gaussian noise, but for many practical applications this is generally supplied by the scope itself and the noise inherent in normal signals.

The resolution enhancement feature uses a flat moving-average filter. This acts as a low-pass filter with good step response characteristics and a very slow roll-off from the pass-band to the stop-band.

Some side-effects will be observed when using resolution enhancement. These are normal and can be counteracted by reducing the amount of enhancement used, increasing the number of samples captured or changing the timebase. Trial and error is usually the best way to find the optimum resolution enhancement for your application. The side-effects include:

- Widened and flattened impulses (spikes)
- Vertical edges (such as those of square waves) turned into straight-line slopes
- Inversion of the signal (sometimes making it look as if the trigger point is on the wrong edge)
- A flat line (when there are not enough samples in the waveform)

Procedure

- Click the **Channel Options** button in the [Channel Setup toolbar](#).
- Use the **Resolution Enhancement** control in the [Channel Options menu](#) to select the effective number of bits, which can be equal to or greater than the [vertical resolution](#) of your scope device.

Quantifying Resolution Enhancement

The table below shows the size of the moving-average filter for each resolution enhancement setting. A bigger filter size requires a higher sampling rate to represent a given signal without significant side-effects (as detailed above).

| Resolution enhancement e (bits) | Number of values n |
|---|--|
| 0.5 | 2 |
| 1.0 | 4 |
| 1.5 | 8 |
| 2.0 | 16 |
| 2.5 | 32 |
| 3.0 | 64 |
| 3.5 | 128 |
| 4.0 | 256 |

Example. Your scope device is a PicoScope 5204 (resolution = 8 bits). You have selected an effective resolution of 9.5 bits. The resolution enhancement is therefore:

$$e = 9.5 - 8.0 = 1.5 \text{ bits.}$$

The table shows that this is achieved using a moving average of:

$$n = 8 \text{ samples.}$$

This number gives a guide to what sort of filtering effect the resolution enhancement will have on the signal. The best way of seeing the actual low-pass filter effect is to add a spectrum view and look at the shape of the noise floor (try dragging the y-axis upwards to see the noise more clearly).

Related topics

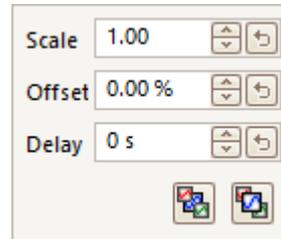
See [Hardware resolution](#) (applies to Flexible Resolution Oscilloscopes only).

7.2.1.2 Axis scaling controls

The **axis scaling controls** are control boxes that let you change the scale and offset of each vertical axis individually. If the axis belongs to a [reference waveform](#) then you can also adjust its delay relative to the live waveforms.



Controls for a live waveform



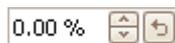
Controls for a reference waveform

There are two ways to open the axis scaling control:

- For any channel displayed in a [view](#): click the colored scaling button () at the bottom of the vertical axis
- For any input channel: Click the [Channel Options button](#) in the [Channels toolbar](#)



Scale control. Increase to magnify the waveform, decrease to reduce it. The vertical axis rescales accordingly so that you can always read the correct voltage from the axis. Click the reset button () to return to a scale of 1.0. The scaling button always shows the selected scale.



Offset control. Increase to move the waveform up the display, decrease to move it down. The vertical axis shifts accordingly so that you can always read the correct voltage from the axis. Adjusting this control is equivalent to clicking and dragging the vertical axis. Click the reset button () to return to an offset of 0.00%.



Delay control (for reference waveforms only). Increase to move the waveform to the left relative to the timing reference point, decrease to move it to the right. Click the reset button () to return to a delay of 0 s.

The location of the timing reference point depends on which [trigger mode](#) PicoScope is in. If the trigger mode is **None** then the delay is measured relative to the left-hand edge of the display. In all other trigger modes, the delay is measured relative to the [trigger](#) marker.



Send to back. Draws the channel behind all others. Use if the channel is obscuring another channel of interest.



Bring to front. Draws the channel in front of all others. Use if the channel is hidden behind another.

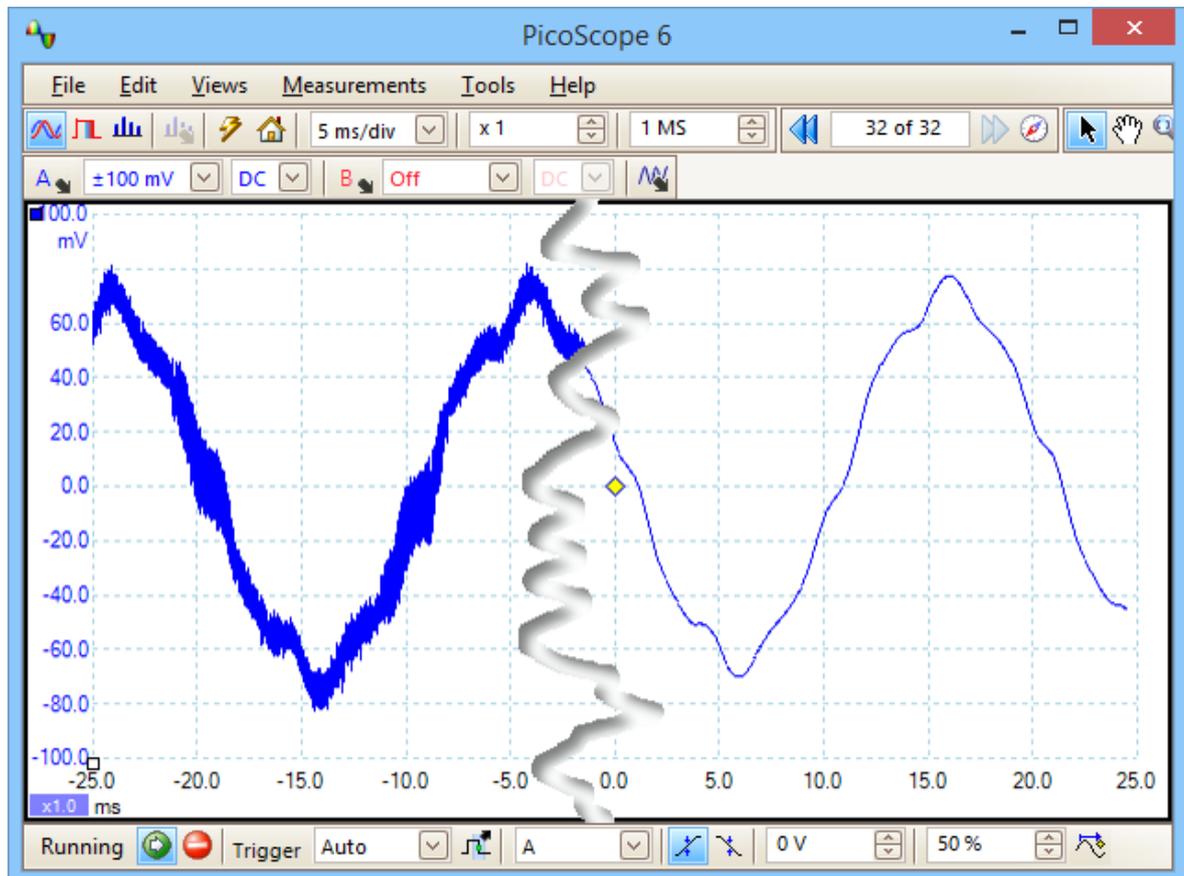
7.2.1.3 Lowpass filtering

The **lowpass filtering** feature can reject high frequencies from any selected input channel. The filtering control is found in the [Advanced Channel Options dialog](#), which is opened by clicking the **Channel Options button** () for the relevant channel on the [Channels toolbar](#). The control determines the cut-off frequency of the filter, which must be below half the sampling rate shown in the [Properties sheet](#).



For availability, see the [Device feature table](#).

Lowpass filtering is useful for rejecting noise. The split screenshot below shows the effect of applying a 1 kHz lowpass filter on a noisy signal. The underlying shape of the signal is preserved but the high-frequency noise is eliminated:



Left: before lowpass filtering. Right: after 1 kHz lowpass filtering.

Filter details

The lowpass filtering algorithm is chosen according to the ratio of the selected cut-off frequency (f_c) to the sampling rate (f_s), as follows:

| $f_c \div f_s$ | Filter type | Description |
|----------------|----------------|---|
| 0.0 to 0.1 | Moving average | A moving average filter is used for low cut-off frequencies. The length of the filter is adjusted to achieve the selected cut-off frequency, which is defined as the first minimum in the frequency response. There is significant signal leakage above the cut-off frequency. This filter changes a vertical edge into a linear slope. |
| 0.1 to < 0.5 | FIR | A finite impulse response filter is used for medium to high cut-off frequencies. This has a monotonic roll-off above the cut-off frequency and therefore suffers from less leakage than the moving average filter. |

You can force PicoScope to use one or other of the filter types by adjusting the **Samples** control in the [Capture Setup toolbar](#) to make the ratio f_c/f_s fall into one of the two ranges shown in the table. As the table shows, the cut-off frequency must be below half the sampling frequency.

7.2.2 ConnectDetect

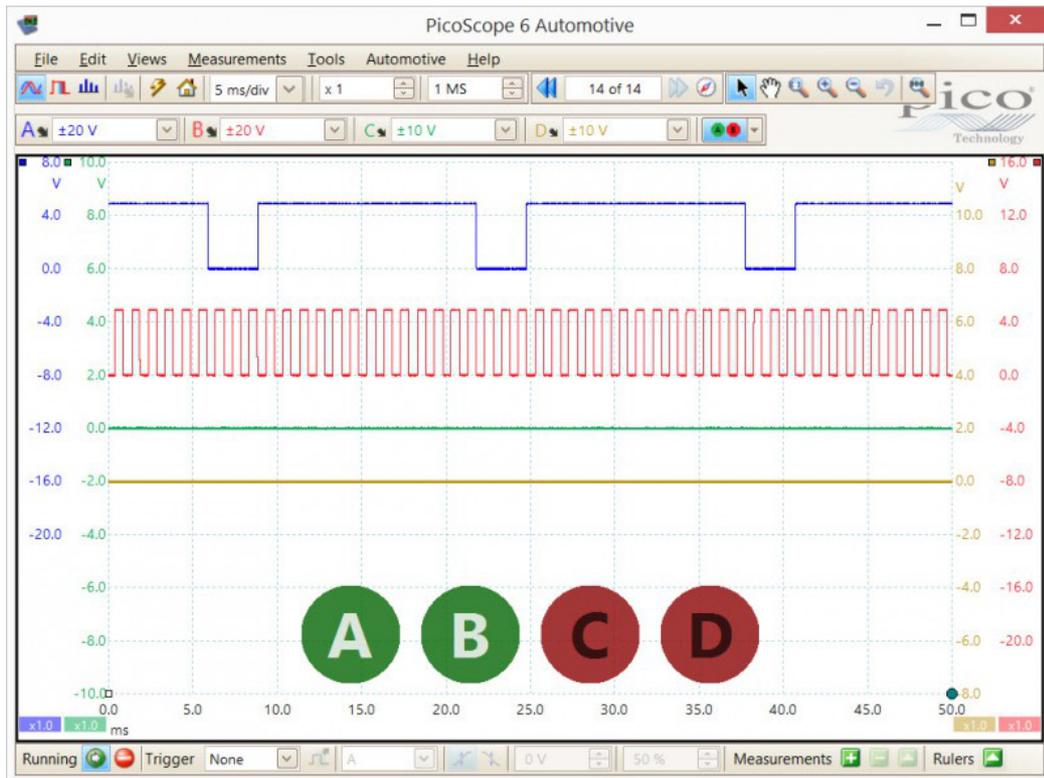
Availability: PicoScope 4225 and 4425 automotive scopes only.
DC coupling mode only.

Purpose: Indicates whether or not a test probe has a good physical connection to the component under test.

Location: To enable ConnectDetect®, click the **ConnectDetect** button  in PicoScope 6 Automotive.

Every channel input on the PicoScope 4225 and 4425 PicoScopes has an LED next to it. When ConnectDetect is activated for a channel, the LED for that channel is either green, to indicate that the test probe is directly connected across a component, or red to indicate that it is not. You can select which channels ConnectDetect is activated for by clicking the arrow on the ConnectDetect button.

An icon representing the LED is also shown on the PicoScope screen, as in the example below, where ConnectDetect has been activated on all four channels, but only Channels A and B are properly connected.



Note that ConnectDetect can only be used with probes that come into direct contact with the component under test. Sensors such as the TA204 coil-on-plug and signal probe, which works using electrical induction from the ignition coils attached to the spark plugs, do not make such a connection, so ConnectDetect will not work on them.

7.2.3 Digital Inputs button

Location: [Channels toolbar](#) ([MSOs](#) only)

Purpose: controls the settings for the digital inputs of a mixed signal oscilloscope ([MSO](#))



Digital on/off. Switches the [digital view](#) on or off. If digital inputs are activated in the [Digital Setup dialog](#), they remain active even when hidden from view.

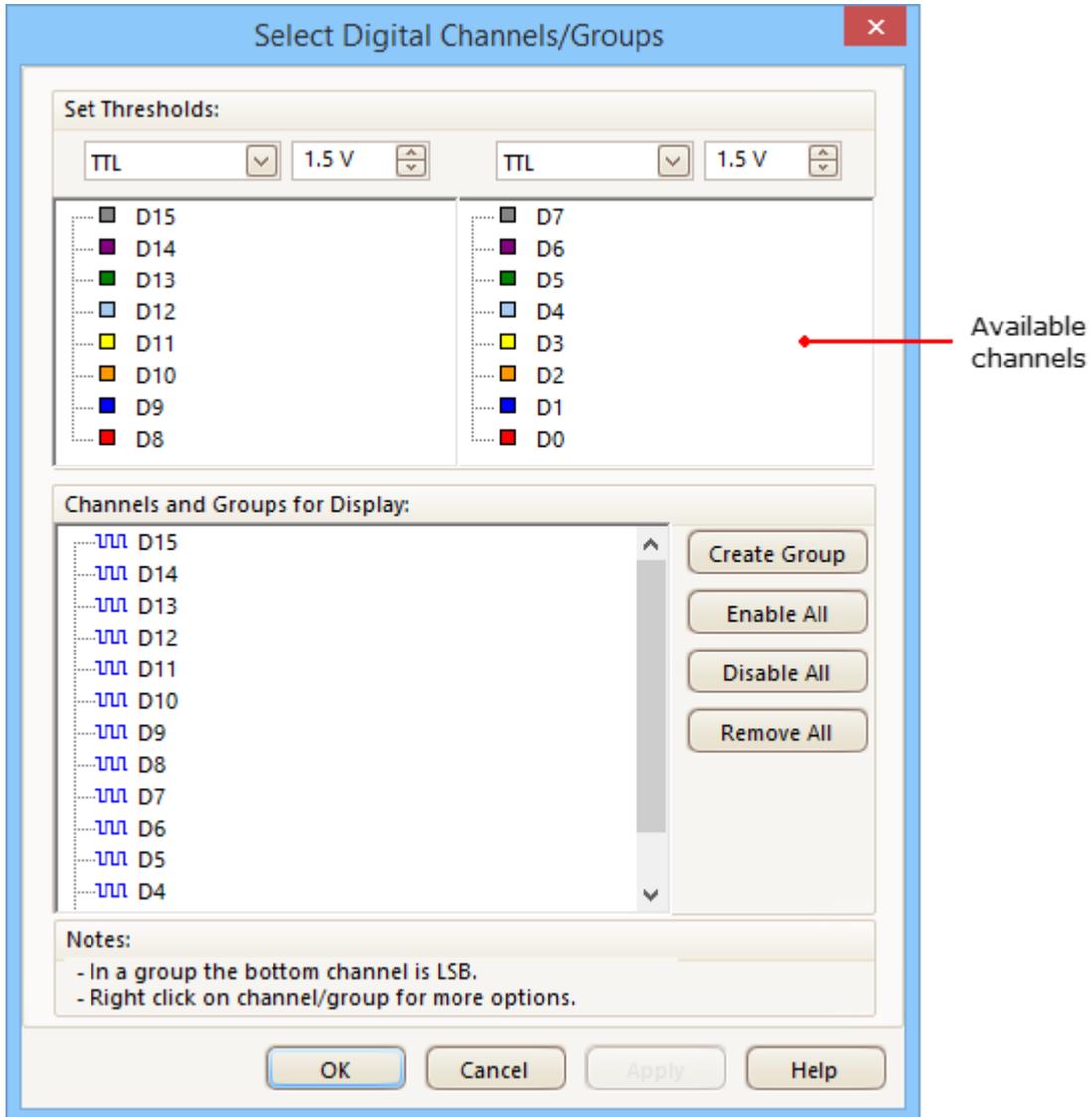


Digital setup. Opens the [Digital Setup dialog](#) for channel selection and options.

7.2.3.1 Digital Setup dialog

Location: [MSO button](#)

Purpose: controls the digital inputs of an MSO (mixed signal oscilloscope)

**Set Thresholds**

Choose the digital threshold voltage from the drop-down list, or select the **Custom** threshold and set your own voltage using the numeric entry control. The preset thresholds are:

| | | | |
|---------------|--------|-----------------|--------|
| TTL | 1.5 V | LVC MOS 1.8 V | 0.9 V |
| CMOS | 2.5 V | LVC MOS 2.5 V | 1.25 V |
| ECL | -1.3 V | LVC MOS 3.3 V | 1.65 V |
| PECL | 3.7 V | LVDS | 100 mV |
| LVPECL | 2 V | 0V Differential | 0 V |
| LVC MOS 1.5 V | 750 mV | | |

Each port has its own independent threshold. Port 0 contains channels D7...D0 and Port 1 contains channels D15...D8.

Available Channels

This section lists the available digital input channels. They will not be displayed unless you add them to the **Channels and Groups for Display** section of the dialog. Click and drag individual channels into the Channels and Groups for Display section, select a range of channels and drag them all at once, or double-click a channel to add it directly.

Channels and Groups for Display

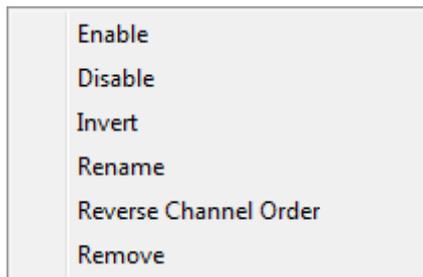
This section lists the digital channels that have been selected for display. Any groups of channels that you have defined will also be listed here.

 indicates a digital channel.

 indicates a group of digital channels. By default, channels added to a group are placed with the most significant bit at the top of the list.

Click the **Create Group** button to add a new group. This adds a new group to the Channels and Groups for Display section, and you can add channels to it by dragging them on top of it: you can do this from the available channels or from the Channels and Groups for Display section.

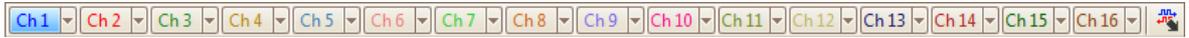
To rename a channel or group, click the name and type. For other operations, right-click the channel or group for a menu of actions:



| | |
|------------------------------|--|
| Enable | Show the channel. All channels in the list are enabled by default. |
| Disable | Hide the channel from display. |
| Invert | Reverse the polarity of this channel. Useful for active-low signals. |
| Rename | Type a new name for the channel. |
| Reverse Channel Order | (Groups only) Reverse the order of channels in the group. |
| Remove | Remove channel from the list. |

7.3 PicoLog 1000 Series Channels toolbar

The **Channels toolbar** controls the settings for each vertical input [channel](#). The toolbar has a different appearance for PicoLog 1000 Series Data Loggers than for PicoScope oscilloscopes (see [Channels toolbar](#) for the standard version).



Channel control. This control contains two buttons in one rectangular outline. Click the small triangle on the left to open the [Channel Options dialog](#) with options for [probes](#), [resolution enhancement](#), [scaling](#) and filtering. Click the channel name to toggle the channel on or off.

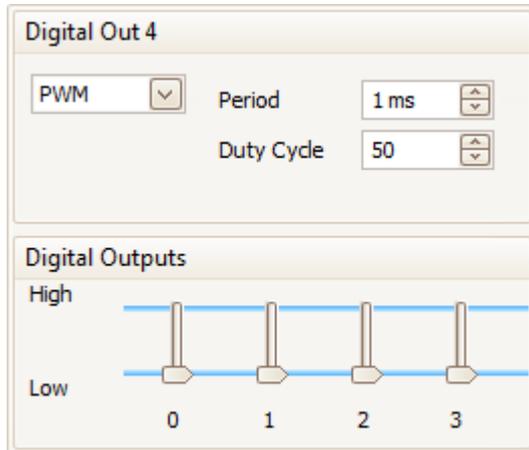


Digital Outputs button. For controlling the 2 or 4 digital outputs of the PicoLog 1000 Series device. Opens the [Digital Outputs dialog](#).

7.3.1 PicoLog 1000 Series Digital Outputs control

Location: **Digital Outputs button**  on the [Channels toolbar](#)

Purpose: controls the [data logger's](#) built-in signal generator



Digital Outputs dialog for the PicoLog 1216

The range of controls available depends on which model of data logger you have.

PWM Output



PWM. The PWM output on some devices can be set to generate a pulse-width modulated waveform. This is a logic signal that is toggled with a specified period and duty cycle. The average value of the signal is proportional to its duty cycle, so it can be processed by an external low-pass filter to produce a signal that is proportional to the duty cycle.

Off: Disable the PWM output.

PWM: Enable the PWM output with the specified controllable **Period** and **Duty Cycle**.

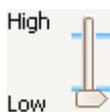


Period. Select the duration of one cycle of the PWM output.

Duty Cycle. The percentage of the PWM signal period that the signal spends at the logic high level. For example, if the period is 1 ms and the duty cycle 25%, then the signal will spend 25% of 1 ms = 250 μ s of each cycle at the logic high level, and the remaining 750 μ s at the logic low level. The voltages of the logic high and low levels are specified in the User's Guide for the data logger, but are typically 0 volts (low) and 3.3 volts (high). Using our example figures, the average value of the PWM output will be 25% x 3.3 volts = 0.825 volts.

Digital Outputs

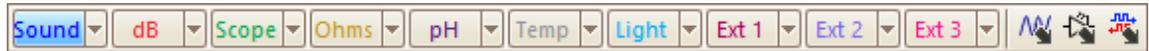
PicoLog PC Data Loggers have one or more digital outputs capable of driving low-current loads.

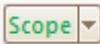
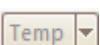
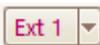


Each output can be set to a high or low logic level by moving the slider.

7.4 USB DrDAQ Channels toolbar

The **Channels toolbar for the USB DrDAQ** controls the settings for each input and output [channel](#):



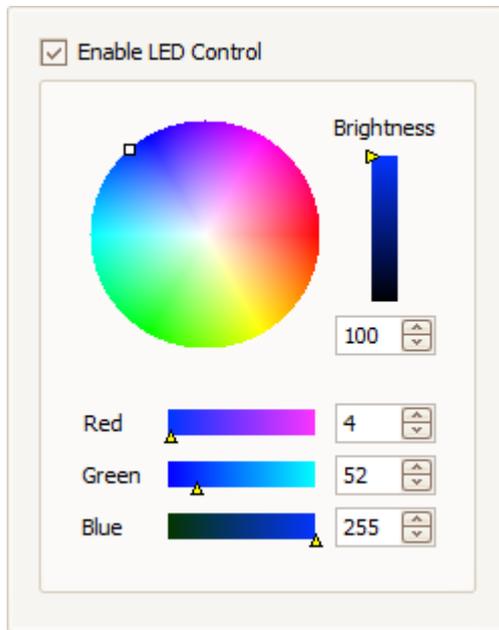
-  **Sound waveform sensor control.** The small arrow sets options for the sound waveform input (measured in uncalibrated amplitude units) using the on-board microphone. Click the channel name to toggle the channel on or off.
-  **Sound level sensor control.** The small arrow sets options for the sound level input (measured in decibels) using the on-board microphone. Click the channel name to toggle the channel on or off.
-  **Scope input control.** The small arrow sets options for the oscilloscope input (the BNC socket marked **Scope**), with options for [probes](#) and [scaling](#). Click the channel name to toggle the channel on or off.
-  **Resistance input control.** The small arrow sets options for the 0 to 1 MΩ resistance-measuring input on the screw-terminal block. Click the channel name to toggle the channel on or off.
-  **pH input control.** The small arrow sets options for the pH and ORP (oxidation/reduction potential) measuring input. Click the channel name to toggle the channel on or off.
-  **Temperature sensor control.** The small arrow sets options for the on-board temperature sensor. Click the channel name to toggle the channel on or off.
-  **Light sensor control.** The small arrow sets options for the on-board light level sensor. Click the channel name to toggle the channel on or off.
- 

 **External sensor controls.** The small arrows set options for external sensor inputs 1 to 3. Click the channel name to toggle the channel on or off.
-  **Signal generator button.** Opens the [Signal Generator dialog](#), which allows you to set the characteristics of the signal generator output.
-  **RGB LED button.** Opens the [RGB LED Control dialog](#), which allows you to set the color of the on-board LED.
-  **Digital Outputs button.** Opens the [Digital Outputs dialog](#), which allows you to control the states of the four digital outputs.

7.4.1 USB DrDAQ RGB LED control

Location: [USB DrDAQ Channels toolbar](#) > **RGB LED button:** 

Purpose: allows you to set the color of the on-board LED to any one of 16.7 million colors



Enable LED Control **Box checked:** you can set the on-board RGB LED to any color

Box clear: the LED has its normal function of flashing to indicate data capture on the input channels

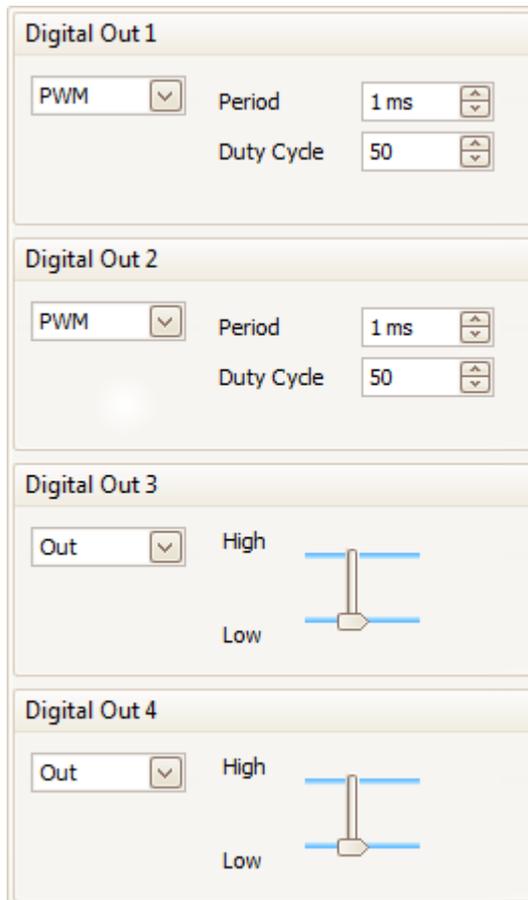
Other controls Experiment with these to see what they do!

7.4.2 USB DrDAQ Digital Outputs control

Location: [USB DrDAQ Channels toolbar](#) > **Digital Outputs button:** 

Purpose: allows you to set the characteristics of the four digital outputs on the screw-terminal block.

Each output has its own set of controls:

**PWM/Out Control**

Set to Out: you can set the output to either a fixed logic low (near 0 V) or a fixed logic high (near 3.3 V)

Set to PWM: the output is a two-level waveform (alternating between 0 V and 3.3 V) with variable **Duty Cycle** and **Period**. The signal can be filtered to produce a DC level proportional to the duty cycle.

Period

the time between successive pulses on the output

Duty Cycle

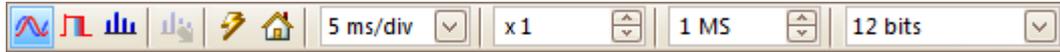
the percentage of the **Period** for which the output is high.

7.5 Capture Setup toolbar

The **Capture Setup toolbar** controls the time-related or frequency-related settings of your oscilloscope.

Scope Mode

In [scope mode](#), the toolbar looks like this:



(See below for different versions of the toolbar in [spectrum mode](#) and [persistence mode](#)).



Scope Mode. Sets up PicoScope to operate as an [oscilloscope](#). Use the **Auto Setup button** to optimize the settings. If you wish, you can add a secondary [spectrum view](#) from the context menu (by right-clicking on the scope view).



Persistence Mode. Toggles [persistence mode](#), which allows old traces to remain on the screen in faded colors while new traces are drawn on top in brighter colors. The use of colors is controlled by the **Persistence Options dialog**. PicoScope will remember any views that were open, so that you can return to them by clicking the **Persistence Mode** button again.



Spectrum Mode. Sets up PicoScope to operate in [spectrum analyzer mode](#). Use the **Auto Setup button** to optimize the settings. If you wish, you can add a secondary [scope view](#) from the context menu (by right-clicking on the scope view).



Auto Setup. Searches for a signal on one of the enabled input channels, then sets up the timebase and signal range to display the signal correctly.



Home. Restores PicoScope to its default settings. Equivalent to the [File > Startup Settings > Load Startup Settings](#) command.



Timebase control. Sets the time represented by a single division of the horizontal axis when the **horizontal zoom control** is set to x1. The timebases available depend on the type of [scope device](#) you are using and, for some devices, on the number and [combination of channels enabled](#) and the selected [trigger mode](#).

Choosing a timebase equal to or longer than PicoScope's *Slow Sampling Transition* setting (200 ms/div by default) causes the scope to switch to a different mode of data transfer. The internal details of this are taken care of by PicoScope, but the slow mode limits the sampling rate to the 'streaming mode sampling rate' specified in the scope device's data sheet. The *Slow Sampling Transition* setting can be changed in the [Tools > Preferences > Sampling](#) dialog.

You can change this control to display the total time across the scope view, rather than the time per division, using the **Collection Time Units** control in the [General](#) page of the [Preferences dialog](#).



Horizontal zoom control. Zooms the view, in the horizontal direction only, by the specified amount. Click the and buttons to adjust the zoom factor, or the button to reset.



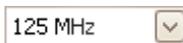
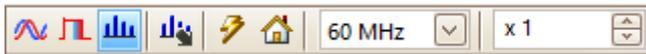
Samples control. Sets the maximum number of samples that will be captured for each channel. If this is larger than the number of pixels across the scope view, you can zoom in to see more detail. The actual number of samples captured is displayed on the [Properties sheet](#), and may be different from the number requested here, depending on which timebase is selected and which scope device is in use. To capture one waveform that occupies the entire buffer memory, first set the [Trigger Mode](#) control to **Single**.



Hardware resolution ([Flexible Resolution Oscilloscopes](#) only). Sets the number of hardware bits used for sampling. The range of options depends on the number of channels enabled and the sampling rate selected. **Automatic Resolution** chooses the highest resolution compatible with the currently selected sampling rate and capture size. To find out what resolution is actually in use, view the **Effective Resolution** value in the [Properties Sheet](#). Resolution can be further increased by software filtering: see [resolution enhancement](#).

Spectrum Mode

In [spectrum mode](#), the **Capture Setup toolbar** looks like this:



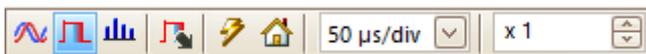
Spectrum Range control. Sets the frequency range across the horizontal axis of the spectrum analyzer when the **horizontal zoom control** is set to x1.



Spectrum Options. Appears if a [spectrum view](#) is open, regardless of whether [scope mode](#) or [spectrum mode](#) is selected. It opens the [Spectrum Options dialog](#).

Persistence Mode

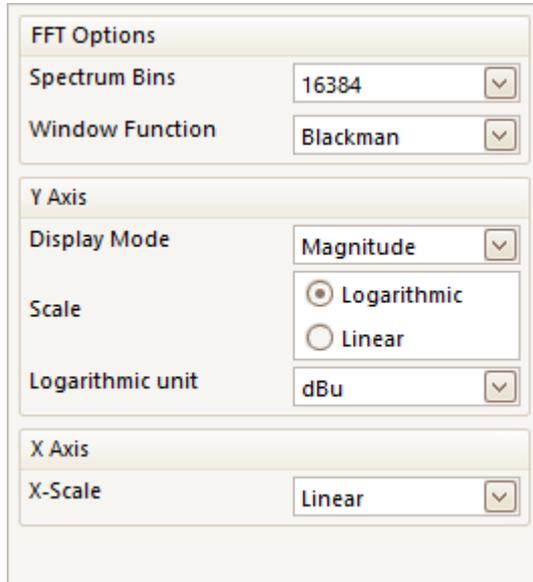
In [persistence mode](#), the **Capture Setup toolbar** looks like this:



Persistence Options. Opens the [Persistence Options dialog](#), which controls several parameters affecting the way that PicoScope represents old and new data in persistence mode.

7.5.1 Spectrum Options dialog

This dialog appears when you click the **Spectrum Options** button in the [Capture Setup toolbar](#). It is available only when a [spectrum view](#) is open. It contains controls that determine how PicoScope converts the source waveform in the current scope view to a spectrum view.



Spectrum Bins

The number of frequency bins into which the spectrum is divided. This control sets the maximum number of frequency bins, which the software may or may not be able to provide depending on other settings. The main constraint is that the number of bins cannot greatly exceed half the number of samples in the source waveform.

If the source waveform contains fewer samples than required (that is, fewer than twice the number of frequency bins), then PicoScope zero-pads the waveform up to the next power of two. For example, if the scope view contains 10,000 samples, and you set Spectrum Bins to 16384, then PicoScope zero-pads the waveform to 16,384 samples, which is the nearest power of two above 10,000. It then uses these 16,384 samples to provide 8,192 frequency bins, not the 16,384 requested.

If the source waveform contains more samples than required, then PicoScope uses as many samples as necessary, starting from the beginning of the waveform buffer. For example, if the source waveform contains 100,000 samples and you request 16,384 frequency bins, PicoScope needs only $2 \times 16,384 = 32,768$ samples, so it uses the first 32,768 samples from the waveform buffer and ignores the rest. The amount of data actually used is displayed as the **Time Gate** setting in the [Properties sheet](#).

Window Function

Allows you to choose one of the standard window functions to reduce the effect of operating on a time-limited waveform. See [Window functions](#).

Display Mode

You can choose **Magnitude**, **Average** or **Peak Hold**.

Magnitude: the spectrum view shows the frequency spectrum of the last waveform captured, whether live or stored in the [waveform buffer](#).

Average: the spectrum view shows a rolling average of spectra calculated from all the waveforms in the [waveform buffer](#). This has the effect of reducing the noise visible in the spectrum view. To clear the averaged data, click [Stop](#) and then [Start](#), or change from **Average** mode to **Magnitude** mode.

Peak Hold: the spectrum view shows a rolling maximum of the spectra calculated from all the waveforms in the buffer. In this mode, the amplitude of any frequency band in the spectrum view will either stay the same or increase, but never decrease, over time. To clear the peak hold data, click [Stop](#) and then [Start](#), or change from **Peak Hold** mode to **Magnitude** mode.

Note: when you switch to Average or Peak Hold mode, there may be a noticeable delay while PicoScope processes the entire contents of the waveform buffer, which may contain many waveforms, to build up the initial display. If this occurs, a progress bar appears at the bottom of the window to show that PicoScope is busy:

**Scale**

Specifies the labelling and scaling of the vertical (signal) axis. This can be one of the following:

Linear:

The vertical axis is scaled in volts RMS.

Logarithmic:

The vertical axis is scaled in decibels, referred to the level selected below in the **Logarithmic unit** control.

dBV: Reference level is 1 volt.

dBu: Reference level is 1 milliwatt with a load resistance of 600 ohms. This corresponds to a voltage of about 775 mV.

dBm: Reference level is one milliwatt into the specified load impedance. You can enter the load impedance in the box below the **Logarithmic unit** control.

Arbitrary dB: Reference level is an arbitrary voltage, which you can specify in the box below the **Logarithmic unit** control.

X Scale

Specifies the scale of the frequency axis:

Linear: The axis is scaled in equal intervals from DC to the frequency set by the [Spectrum Range](#) control.

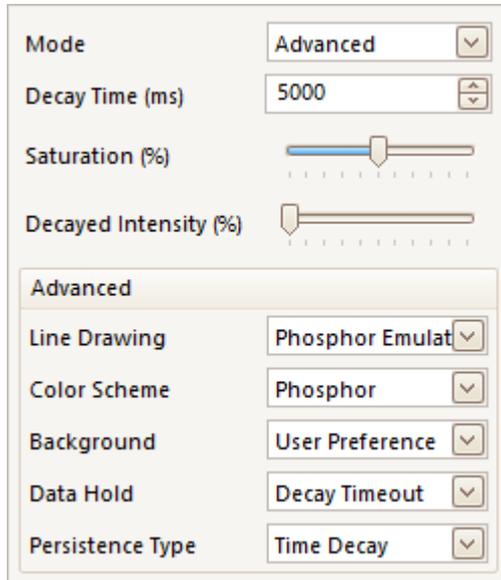
Log 10: The axis is scaled in decades ending at the specified **Spectrum Range** frequency, and beginning at a number of decades below that as specified by the **Number of Decades** control.

Number of Decades

Specifies the number of decades into which the frequency axis is divided when **X Scale** is set to **Log 10**.

7.5.2 Persistence Options dialog

This dialog appears when you click the **Persistence Options** button  in the [Capture Setup toolbar](#). It is available only when [persistence mode](#) is selected. It controls the colors and fading algorithm used to distinguish new or frequent data from old or intermittent data in the persistence view.



Mode **Digital Color.** This mode uses a range of colors to indicate the frequency of waveform data. Red is used for the most frequent data, and less frequent data is represented successively by yellow and blue.

Analog Intensity. This mode uses color intensity to indicate the age of waveform data. The latest data is drawn at full intensity in the selected color for that channel, with older data being represented by paler shades of the same color.

Fast. A mode with simplified display options to enable the fastest possible update rate.

Advanced. This mode opens up a **Custom Options** section at the bottom of the dialog that let you customise the persistence mode display.

Decay Time The time, in milliseconds, taken for waveform data to fade from maximum intensity to minimum intensity or from red to blue. The longer the decay time, the longer the older waveforms will remain on the screen.

Saturation The intensity or color with which new waveforms are drawn.

Decayed Intensity The intensity or color to which the oldest waveforms decay when the decay time expires. If the decayed intensity is zero, then older waveforms will be completely erased from the display after the decay time. For non-zero values of decayed intensity, old waveforms will remain indefinitely on the screen at that intensity unless overwritten by new ones.

Custom Options

These options are available when **Mode** is set to **Advanced**.

Line Drawing The type of line drawn between samples that are adjacent in time.

Phosphor Emulation. Joins each pair of sample points with a line whose intensity varies inversely with the slew rate.

Constant Density. Joins each pair of sample points with a line of uniform color.

Scatter. Draws sample points as unconnected dots.

Color Scheme

Phosphor. Uses a single hue for each channel, with varying intensity.

Color. Uses a color from red to blue to represent the age of each waveform.

Background

Black. Overrides the [Color Preferences dialog](#). This is the default.

White. Overrides the [Color Preferences dialog](#).

User Preference. Sets the background color to the preference set in the [Colors](#) page of the [Preferences dialog](#).

Data Hold

This option is enabled only when **Persistence Mode** (see below) is set to **Time Delay**.

Decay Timeout. Old waveforms fade until they reach **Decayed Intensity** and then disappear.

Infinite. Old waveforms fade until they reach **Decayed Intensity** and then remain indefinitely unless overwritten by new waveforms.

Persistence Mode

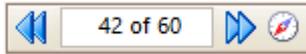
Fast. The **Line Drawing**, **Decay Time**, **Saturation** and **Decayed Intensity** options are disabled to maximize the update rate. In addition, [resolution enhancement](#), [lowpass filtering](#), [sin\(x\)/x interpolation](#) and nonlinear [custom probes](#) are unavailable in this mode. This mode requires a scope that supports [Rapid Triggering](#) (see [Device Features](#) table).

Time Delay. Points on the display are drawn at full intensity when hit by a waveform, and are then allowed to decay to **Decayed Intensity**. The behavior after this depends on the **Data Hold** setting (see above).

Frequency. Points on the display are drawn with a color or intensity that depends on the frequency with which they are hit by waveforms.

7.6 Buffer Navigation toolbar

The **Buffer Navigation toolbar** allows you to select a waveform from the waveform buffer.



What is the waveform buffer?

Depending on the settings you have chosen, PicoScope may store more than one waveform in its waveform buffer. When you click the [Start](#) button or change a [capture setting](#), PicoScope clears the buffer and then adds a new waveform to it each time the scope device captures data. This continues until the buffer is full or you click the [Stop](#) button. You can limit the number of waveforms in the buffer to a number between 1 and 10,000 using the [General preferences](#) page.

You can review the waveforms stored in the buffer using these buttons:



First waveform button. Display waveform 1.



Previous waveform button. Display the previous waveform in the buffer.

42 of 60

Waveform number indicator. Show which waveform is currently displayed, and how many waveforms the buffer holds. You can edit the number in the box and press **Enter**, and PicoScope will jump to the specified waveform.



Next waveform button. Display the next waveform in the buffer.



Last waveform button. Display the last waveform in the buffer.



Buffer Overview button. Open the [Buffer Overview window](#) for quick selection of buffer waveforms.

7.7 Measurements toolbar

The **Measurements toolbar** controls the [measurements table](#).



It contains the following buttons:

- | | |
|---------------------------|---|
| Add Measurement | Opens the Add Measurement dialog , allowing you to add a measurement of a waveform to the selected view. |
| Delete Measurement | Deletes the currently selected row from the measurements table . |
| Edit Measurement | Opens the Edit Measurement dialog for the currently selected measurement. You can also edit a measurement by double-clicking on a row of the measurements table . |

This toolbar normally resides at the bottom of the program window, but can be moved to the top using the [Tools > Preferences > Options > Bottom toolbar at top](#) control.

7.8 Signal Generator button

The **Signal Generator button**  allows you to set up your [scope's](#) test signal generator, if it has one, or the demo signal settings if PicoScope is in [demo mode](#).

If your scope has a built-in signal generator then clicking the **Signal Generator button** opens the [Signal Generator dialog](#).

If PicoScope is in [demo mode](#) then clicking the **Signal Generator button** opens the [Demo Signals menu](#).

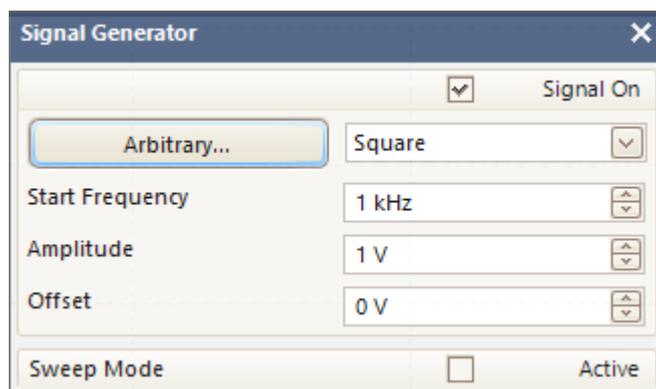
7.8.1 Signal Generator dialog (PicoScope devices)

Location: **Signal Generator button**  on the toolbar

Purpose: controls the [scope device's](#) built-in signal generator

Not all scope devices have a signal generator, and those that do have a varying range of controls in the signal generator dialog. See the [Device feature table](#) for details.

7.8.1.1 Basic controls



Signal On. Check this box to enable the signal generator.

Signal Type. Select the type of signal to be generated. The [list of signal types](#) depends on the capabilities of the scope device.

Import. Opens a file selection dialog that allows you to import an [arbitrary waveform file](#). The file will be loaded into the [arbitrary waveform generator](#) and the generator switched on. This button is available only if your scope has an [arbitrary waveform generator](#).

Arbitrary. Opens the [Arbitrary Waveform window](#). This button is available only if your scope has an [arbitrary waveform generator](#).

Start Frequency. Type in this box or use the spin buttons to select the frequency. If the scope device has a frequency sweep generator, then this box sets the start frequency of the sweep.

Amplitude. The amplitude of the waveform measured from peak to peak. For example, if **Amplitude** is 1 V and **Offset** is 0 V, the output will have a negative peak of -0.5 V and a positive peak of +0.5 V.

Offset. The mean value of the signal. For example, when **Offset** is 0 V, a sine or square wave will have equal positive and negative peak voltages.

7.8.1.2 Sweep controls

The signal generator normally generates a fixed frequency set by the **Start Frequency Control**. In sweep mode, it generates a frequency that varies between two specified limits.

Active. Check this box to enable sweep mode.

Sweep Type. Specifies the direction in which the frequency is swept.

Stop Frequency. In sweep mode, the generator stops increasing the frequency when it reaches the **Stop Frequency**.

Frequency Increment. In sweep mode, the generator increases or decreases the frequency by this amount every **Increment Time Interval**.

Increment Time Interval. In sweep mode, the generator increases or decreases the frequency by **Frequency Increment** each time this interval ends.

7.8.1.3 Trigger controls

The signal generator normally runs continually. If you enable triggering, the signal generator waits for a specified event before generating an output.

Active. Check this box to enable signal generator triggering.

Trigger Source. Specifies the signal that will be used to trigger the signal generator:

Scope. The same trigger condition that triggers the scope.

Manual. The **Trigger Now** button or an Alarm*.

Ext Input. The trigger input marked **EXT** (if fitted) on the scope device.

Aux Input. The trigger input marked **AUX** (if fitted) on the scope device.

* An **Alarm** can also be made to trigger the signal generator. The Trigger Source must be set to **Manual** for this operation. Other configurations are available in the [Alarms dialog](#).

Type. The condition that must be met by the trigger signal:

Rising. The signal generator starts running when the trigger signal goes from low to high.

Falling. The signal generator starts running when the trigger signal goes from high to low.

Gate High. The signal generator runs whenever the trigger signal is high.

Gate Low. The signal generator runs whenever the trigger signal is high.

Cycles per Trigger. The number of cycles of the specified waveform to generate each time the generator is triggered. If the trigger **Type** is **Gate High** or **Gate Low**, the generator will stop when the gate signal goes inactive even if the requested number of cycles has not been generated.

Threshold. Available only when **Trigger Source** is **Ext Input**. Sets the voltage level used to distinguish between the high and low states of the trigger signal.

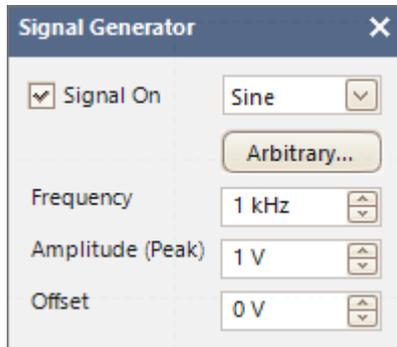
Manual Trigger. Available only when **Trigger Source** is **Manual**. Triggers the signal generator to produce the specified number of cycles (or sweeps, if the signal generator is in [sweep mode](#)).

7.8.2 Signal Generator dialog (USB DrDAQ)

Location:

[Signal Generator button](#)  on the [USB DrDAQ Channels toolbar](#)

Purpose: controls the USB DrDAQ's built-in signal generator



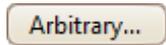
Signal generator dialog for the USB DrDAQ

Basic controls

Signal On. Check this box to enable the signal generator.



Signal Type. Select the shape of the waveform to be generated.



Arbitrary. Opens the [Arbitrary Waveform window](#), allowing you to define your own waveform shape.



Frequency. Type in this box or use the spin buttons to select the frequency of the output waveform.



Amplitude. The amplitude of the waveform measured from peak to peak. For example, if **Amplitude** is 1 V and **Offset** is 0 V, the output will have a negative peak of -0.5 V and a positive peak of +0.5 V.

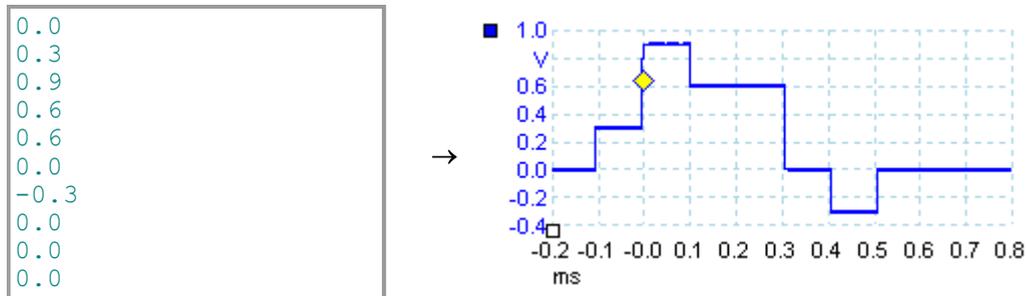


Offset. The mean value of the signal. For example, when **Offset** is 0 V, a sine or square wave will have equal positive and negative peak voltages.

7.8.3 Arbitrary waveform files

Some PicoScope PC Oscilloscopes have an [arbitrary waveform generator](#) (AWG), which is enabled using the [Signal Generator dialog](#). PicoScope can program the AWG with a standard waveform, such as a sine or square wave, or an arbitrary waveform that you create or import from a text file.

A text file for PicoScope 6 is a list of decimal floating-point values, as in this example:



The file may have between 10 and 8,192 values, as many it needs to define the waveform. Each line may have more than one value, in which case the values must be separated by tabs or commas.

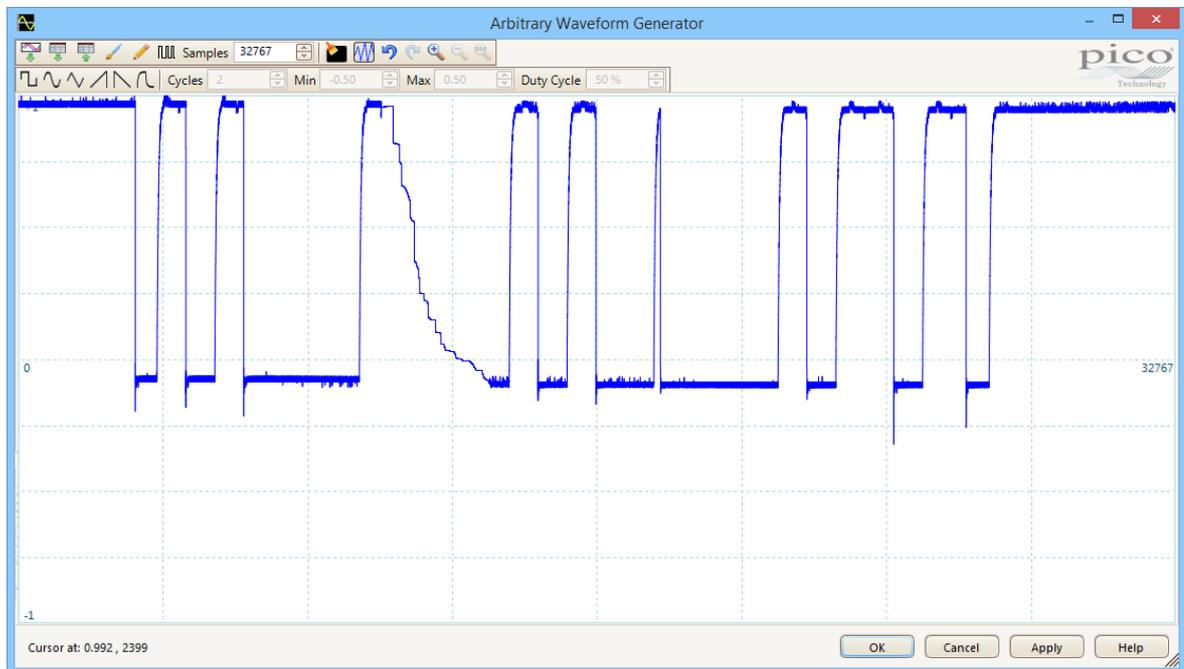
The values are samples between -1.0 and +1.0 and must be equally spaced in time. The output is scaled to the amplitude selected in the [Signal Generator dialog](#), and the selected offset is added if necessary. For example, if the signal generator amplitude is set to 1 V and the offset to 0 V, then a sample value of -1.0 corresponds to an output of -1.0 V and a sample of +1.0 corresponds to an output of +1.0 V.

The file should contain exactly one cycle of the waveform, which will then be played back at the speed specified in the [Signal Generator dialog](#). In the example above, the signal generator was set to 1 kHz, so one cycle of the waveform lasts for 1 ms. There are 10 samples in the waveform, so each sample lasts for 0.1 ms.

7.8.4 Arbitrary Waveform Generator window

Location: [Signal Generator dialog](#) > **Arbitrary**

Purpose: allows you to import, edit, draw and export arbitrary waveforms to load into your scope's [arbitrary waveform generator](#). You can also import and export the data in [CSV format](#) for use in other applications.



Once the desired waveform appears in the window, click **OK** or **Apply** to start using it.

Toolbar buttons



Import from channel. Opens the [Import from Channel dialog](#), which lets you copy a waveform from the scope into the arbitrary waveform window.



Import. Displays an **Open** dialog to allow you to import an arbitrary waveform from a [text file](#).



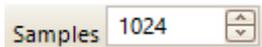
Export. Displays a **Save As** dialog to allow you to save the arbitrary waveform as a [text file](#).



Freehand drawing. Enters freehand drawing mode, in which you can draw any waveform shape using the mouse.



Straight-line drawing. Enters straight line mode, in which you can click on the waveform to draw a straight line from the previous point. To start a new series of lines, click the button again.



Samples. The number of samples in the arbitrary waveform. Each sample represents the signal value at a given instant in time, and the samples are equally spaced in time. For example, if there are 1024 samples and the [arbitrary waveform generator](#) is set to play back at 1 kHz, then each sample represents $(1/1 \text{ kHz} \div 1024)$ or about 0.98 microseconds.



Bit stream. Draws a sequence of bits according to binary or hex data that you specify. The logic high and low levels are adjustable.



Clear. Deletes the arbitrary waveform.



Normalize. Adjusts the waveform vertically so that it occupies the full [-1, +1] range.



Undo and Redo. The **Undo button** reverses the last change made to the arbitrary waveform. The **Redo button** reverses the last action of the **Undo button**.



Zoom tools. To zoom the time axis in or out, click the **+** or **-** zoom button and then click on the waveform area. Click the **100%** button to restore the time axis to its original scale.

Waveform settings



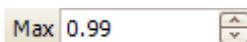
Standard waveform shapes. Draw a standard waveform with the settings specified in the numerical controls below the toolbar. The current waveform will be erased.



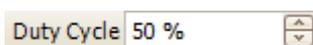
Cycles. The number of cycles to draw. This control is used in conjunction with the **Standard waveform shapes buttons**. Select one of the standard waveform shapes and then enter the number of cycles, and PicoScope will draw the requested number of cycles of the waveform.



Minimum. When one of the **Standard waveform shapes buttons** is pressed, this control sets the minimum signal level.



Maximum. When one of the **Standard waveform shapes buttons** is pressed, this control sets the maximum signal level.



Duty cycle. When a square, triangular or ramp waveform is selected using one of the **Standard waveform shapes buttons**, this control sets the duty cycle of the signal. Duty cycle is defined as the time that the signal spends above zero volts divided by the total cycle time. Thus, a symmetrical square or triangular wave has a duty cycle of 50%. Reducing the duty cycle shortens the positive part of the cycle and lengthens the negative part, and increasing the duty cycle does the opposite.

Other buttons**OK**

Copies the waveform from the graphical editor into the arbitrary waveform generator and returns to the main [PicoScope](#) window.

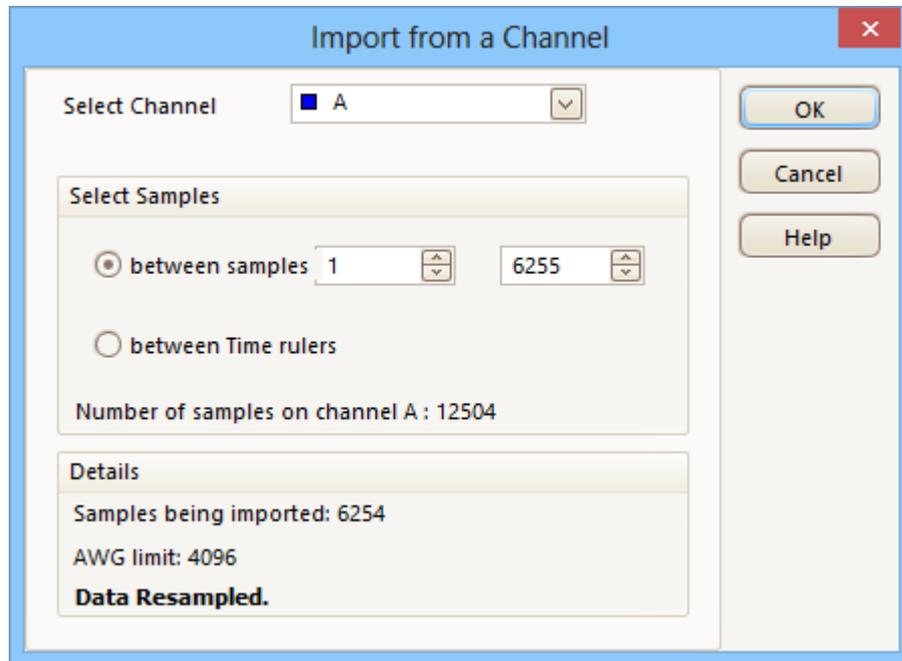
Apply

Copies the waveform from the graphical editor into the arbitrary waveform generator and remains in the **Arbitrary Waveform Generator window**.

7.8.4.1 Import from a Channel dialog

Location: [Arbitrary Waveform window](#) > **Import from a Channel button** ()

Purpose: allows you to copy captured data from a scope channel to the [Arbitrary Waveform window](#)



Select Channel You can import the latest waveform from any available channel.

Select Samples By default, the entire capture is imported. This control allows you to specify a subset of the capture, either between specified sample numbers or between rulers. The subset will be scaled to fit the number of samples specified in the **Samples** control in the [Arbitrary Waveform window](#).

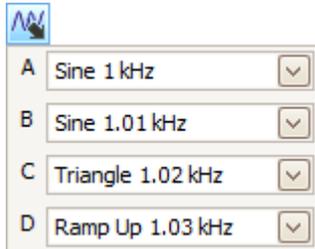
7.8.5 Demo Signals menu

Location: start PicoScope with no scope device plugged in and select [Demo mode](#),

then click the [Signal Generator button](#) 

Purpose: allows you to set up test signals so that you can experiment with PicoScope when no scope device is connected

The Demo Signals menu looks like this:



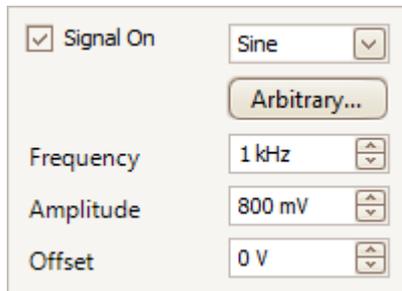
Click one of the channels to open the [Demo Signals dialog](#), which will allow you to set up a signal from that channel.

7.8.6 Demo Signals dialog

Location: start PicoScope with no scope device plugged in and select [Demo mode](#).

- [Signal Generator button](#) ()
- select channel

Purpose: controls one channel of the demo signal source, a feature of PicoScope that creates a variety of test signals to simulate a scope device.


**Signal On**

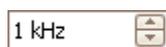
Check this box to enable the demo signal source.

**Signal type**

Select from a list of standard signal types.

**Arbitrary Waveform**

Open the [Arbitrary Waveform Editor](#).

**Frequency**

Type your desired frequency in hertz, or use the spin buttons.

**Amplitude**

Type your desired amplitude in volts, or use the spin buttons.

**Offset**

Enter a number to add a DC offset to the demo signal. By default, the demo signals have a mean value of zero volts.

7.9 Start / Stop toolbar

The **Start / Stop toolbar** allows you to start and stop the [scope device](#). Click anywhere on the toolbar, or press the start/stop key on the keyboard (by default, the space bar), to start or stop sampling.



Start icon. Highlighted if the oscilloscope is sampling.



Stop icon. Highlighted if the oscilloscope is stopped.

This toolbar normally resides at the bottom of the program window, but can be moved to the top using the [Tools > Preferences > Options > Bottom toolbar at top](#) control.

7.10 Triggering toolbar

The **Triggering toolbar** tells the scope device when to start capturing data. See also: [Trigger](#).



Trigger Mode. The list of available modes varies depending on the type of [scope device](#) in use.

None: PicoScope acquires waveforms repeatedly without waiting for a signal to trigger on.

Auto: PicoScope waits for a trigger event before capturing data. If there is no trigger event within a reasonable time, it captures data anyway. It repeats this process until you click the [Stop button](#).

Auto mode does not set the trigger level automatically.

Repeat: PicoScope waits indefinitely for a trigger event before displaying data. It repeats this process until you click the [Stop button](#). If there is no trigger event, PicoScope displays nothing.

Single: PicoScope waits once for a trigger event, then stops sampling. To make PicoScope repeat this process, click the [Start button](#). The **Single** trigger is the only type that allows one capture to fill the entire buffer memory.

Rapid: PicoScope instructs the [scope device](#) to acquire a sequence of waveforms with the minimum possible delay between them. The display is not updated until the last waveform in the sequence has been captured. When the operation is finished, you can step through the waveforms using the [Buffer Navigation toolbar](#).

Note: rapid triggering is available only on certain devices (see [device feature table](#)) and on timebases of 1 ms/div or faster.

ETS: Equivalent Time Sampling. PicoScope captures many cycles of a repetitive signal, then combines the results to produce a single waveform with higher time-resolution than is possible with a single capture. For accurate results, the signal must be perfectly repetitive and the trigger must be stable. ETS is not available on mixed-signal oscilloscopes when digital channels are enabled.

If you select ETS when an [Advanced Trigger](#) type is enabled, the trigger type will revert to **Simple Edge** and the **Advanced Triggering** button will be disabled.



Advanced Triggering. Click to open the [Advanced Triggering dialog](#), which gives you extra trigger types beyond the simple edge trigger. If this button is disabled, it is because either **None** or **ETS** is selected in the trigger mode control, or your scope device does not support this mode. To enable the **Advanced Triggering** button, set the control to another trigger mode, such as **Auto**, **Repeat** or **Single**.



Trigger Source. This is the channel that PicoScope monitors for the [trigger](#) condition.



Rising Edge. Click to trigger on the rising edge of the waveform.



Falling Edge. Click to trigger on the falling edge of the waveform.



Trigger Level. Sets the [trigger](#) level. You can also set the trigger level by dragging the [trigger marker](#) up or down on the screen.



Pre-trigger Time (0% to 100%). This parameter controls how much of the waveform appears before the trigger point. It defaults to 50%, which puts the [trigger marker](#) in the middle of the screen. You can also control this parameter by dragging the [trigger marker](#) to the left or right.

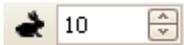


Post-trigger Delay Enable. Click this button to toggle the **Post-trigger Delay control** (see next item).



Post-trigger Delay. The post-trigger delay is the time that PicoScope waits after the trigger point before sampling. You can also modify this parameter by dragging the [trigger marker](#) while the **Post-trigger Delay button** is enabled. As you drag the marker, you will see the [post-trigger arrow](#) displayed briefly. For this control to have an effect, you must first make sure that the **Post-trigger Delay button** is enabled.

See the reference topic [Trigger Timing](#) for information on how the Pre-trigger Time and Post-trigger Delay controls interact.



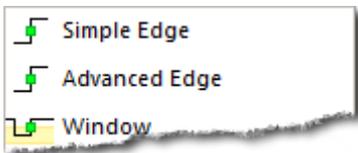
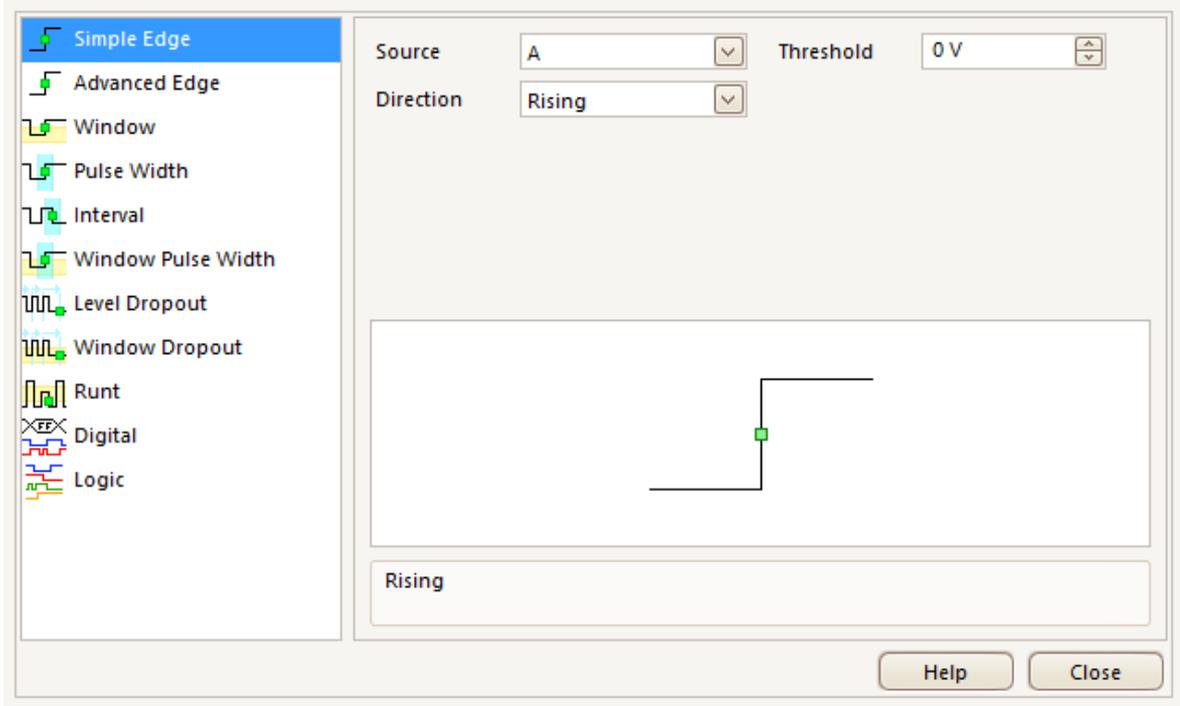
Rapid Captures. In [Rapid](#) trigger mode, this is the number of waveforms to capture in a sequence. They will be captured with the minimum possible [dead time](#) between them.

This toolbar normally resides at the bottom of the program window, but can be moved to the top using the **Move Trigger toolbar to top** control in [Tools > Preferences > Options](#).

7.10.1 Advanced Triggering dialog

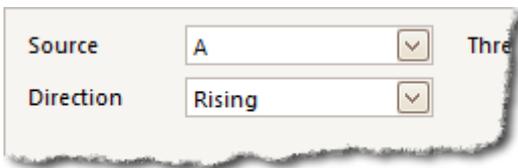
Location: [Triggering toolbar](#) > **Advanced Triggering button** ()

Purpose: allows you to set up more complex trigger types than simple edge-triggering



Advanced trigger types list. This control lists all the available [advanced trigger types](#). Click on the condition you require, and a diagram and description will appear on the right of the dialog.

If [ETS triggering](#) is enabled in the [triggering toolbar](#), the trigger type will be set to **Simple Edge**.



Advanced triggering options. The options available depend on the trigger type selected. See [Advanced trigger types](#). Instructions and diagrams also appear in the dialog.

7.10.2 Advanced trigger types

The **advanced trigger types** can be switched on in the [Advanced Triggering dialog](#).

For all trigger types except [Digital](#), the first step is to select which signal the scope should use as the trigger; so set **Source** to either **A**, **B**, **Ext** or **AuxIO**. These names correspond to the BNC input connectors on the scope device. Then choose one of the trigger types below.

 **Simple Edge.** This type provides the same **Rising** and **Falling** edge triggers that are available from the [Triggering toolbar](#). It is included in this dialog as an alternative method of setting up the Simple Edge trigger.

You can set the trigger **Threshold** while in the **advanced triggering dialog**, or alternatively you can drag the [Trigger marker](#) on the scope view.

This is the only trigger type that is compatible with [ETS](#) mode.

 **Advanced Edge.** This trigger type adds an extra **Rising or Falling** edge trigger, and Hysteresis, to the **Simple Edge** trigger. The **Rising or Falling** option triggers on both edges of a waveform, and is useful for monitoring pulses of both polarities at once (dual-edge triggering). [Hysteresis](#) is described in a separate topic.

 **Window.** This trigger type detects when the signal enters or leaves a specified voltage window. The **Direction** control specifies whether the trigger should detect the signal entering the window, leaving it, or both. **Threshold 1** and **Threshold 2** are the upper and lower voltage limits of the window. The order in which you specify the two voltages does not matter. [Hysteresis](#) can be set to reduce the number of false triggers on a noisy signal, and is described in a separate topic.

 **Pulse Width.** This trigger type detects pulses of a specified width.

First set the **Pulse Direction** to either **Positive** or **Negative** according to the polarity of the pulse you are interested in.

Next, set one of the four **Condition** options:

Greater than triggers on pulses wider than the specified time.

Less than triggers on pulses that are narrower (useful for finding glitches).

Inside time range triggers on pulses that are wider than **Time 1** but no wider than **Time 2** (useful for finding pulses that meet a specification).

Outside time range does the opposite: it triggers on pulses that are either narrower than **Time 1** or wider than **Time 2** (useful for finding pulses that violate a specification).

Next, set the trigger **Threshold** in volts or other units, or drag the [Trigger marker](#) on the scope view.

Finally, set up **Time 1** (and **Time 2** if present) to define the pulse width.



Interval. This type lets you search for two successive edges of the same polarity that are separated by a specified interval of time.

First, set the **Starting edge** to either **Rising** or **Falling** according to the polarity of the edges you are interested in.

Next, select one of the four **Condition** options:

Greater than triggers when the second edge occurs later than **Time 1** after the first edge (useful for detecting missing events).

Less than triggers when the second edge occurs earlier than **Time 1** after the first edge (useful for detecting timing violations and spurious edges).

Inside time range triggers when the second edge is later than **Time 1** after the first edge and earlier than **Time 2** (useful for finding valid edges).

Outside time range triggers when the second edge is earlier than **Time 1** after the first edge or later than **Time 2** (useful for finding spurious edges).

Finally, set up **Time 1** (and **Time 2** if present) to define the time interval.



Window pulse width. This is a combination of the window trigger and the pulse width trigger. It detects when the signal enters or leaves a voltage range for a specified period of time.



Level dropout. This detects an edge followed by a specified time with no edges. It is useful for triggering on the end of a pulse train.



Window dropout. This is a combination of the window trigger and the dropout trigger. It detects when the signal enters a specified voltage range and stays there for a specified time. This is useful for detecting when a signal gets stuck at a particular voltage.



Runt. Detects a pulse that crosses one threshold and then falls below the same threshold, without crossing the second threshold. This is typically used for finding pulses that fail to reach a valid logic level.



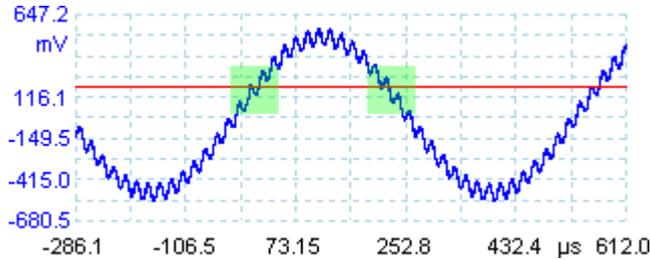
Digital. ([MSO](#) devices only) Triggers on a combination of the state of the digital inputs and a transition (edge) on one digital input. See [Digital trigger](#).



Logic. This detects a logical combination of the scope's inputs. The conditions that can be applied to each input vary: analog inputs can be edge-, level- or window-qualified; EXT and D15...D0 (if present) are level-qualified with a variable threshold; and AUX is level-qualified with a fixed TTL threshold. See [Logic trigger](#).

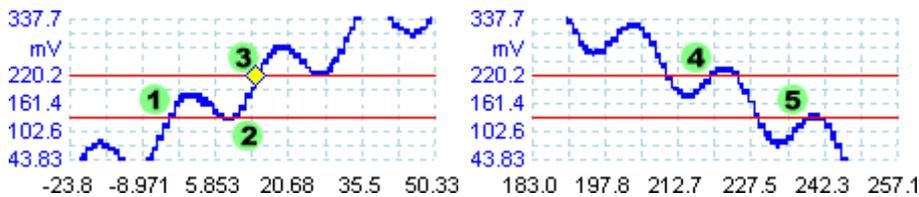
7.10.2.1 Hysteresis

Hysteresis is a feature of the [advanced trigger types](#) in PicoScope 6 that reduces false triggering on noisy signals. When hysteresis is enabled, a second trigger threshold voltage is used in addition to the main trigger threshold. The trigger fires only when the signal crosses the two thresholds in the correct order. The first threshold arms the trigger, and the second causes it to fire. An example will help to illustrate how this works.



Noisy signal with a single threshold

Consider the very noisy signal above. It is difficult to trigger reliably on this signal with a normal rising edge trigger because it crosses the trigger threshold, the red line in this picture, several times in one cycle. If we zoom in on the highlighted parts of the signal, we will see how hysteresis can help.



Noisy signal with hysteresis threshold

In these zoomed-in views, the original threshold is the lower red line. The upper red line is the second threshold used by the hysteresis trigger.

The signal rises across the lower threshold at (1) and (2), arming the trigger but not firing it. At (3) the signal finally crosses the upper threshold, firing the trigger. On the falling edge of the signal, at (4) and (5), rising edges of noise pulses cause the signal to cross the upper and lower thresholds, but in the wrong order, so the trigger is not armed and does not fire. Thus the trigger occurs at only one well-defined point in the cycle (3), despite the noise on the signal.

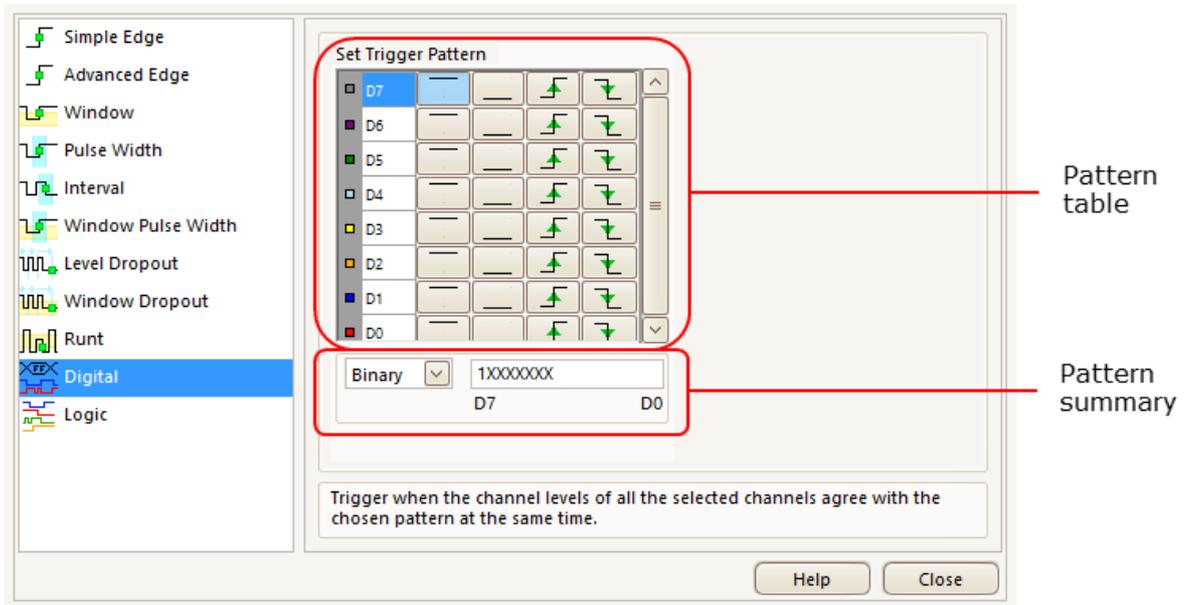
Hysteresis is enabled by default for all the advanced trigger types. The **Hysteresis** controls in the [Advanced triggering dialog](#) let you change the hysteresis voltage as a percentage of full scale. The trigger marker  shows the size of the hysteresis window.

7.10.2.2 Digital trigger dialog

Location: [Advanced triggering dialog](#) > [Digital](#)  and [Logic](#)  buttons

Purpose: sets up triggering on digital inputs

Applicability: [MSO devices](#) only



Pattern table

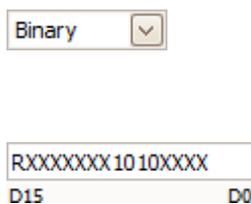
Lists all available inputs as selected in the [Digital Setup dialog](#). Each one can be monitored for a low or high level or a rising or falling edge, or ignored. Any number of levels can be specified, but no more than one transition (edge).

| | | | | | |
|--------------------------|----|-----|-----|--|--|
| <input type="checkbox"/> | D7 | --- | --- | | |
| <input type="checkbox"/> | D7 | --- | --- | | |
| <input type="checkbox"/> | D7 | --- | --- | | |
| <input type="checkbox"/> | D7 | --- | --- | | |
| <input type="checkbox"/> | D7 | --- | --- | | |

- D7 = X (don't care)
- D7 = 0 (low level)
- D7 = 1 (high level)
- D7 = R (rising edge)
- D7 = F (falling edge)

Pattern summary

This section contains the same settings as the **pattern table** but in a more concise format.



The numerical format to use for this section: **Binary** or **Hex(adecimal)**.

The complete triggering pattern and transition. In

Binary mode the bits are labeled as follows:

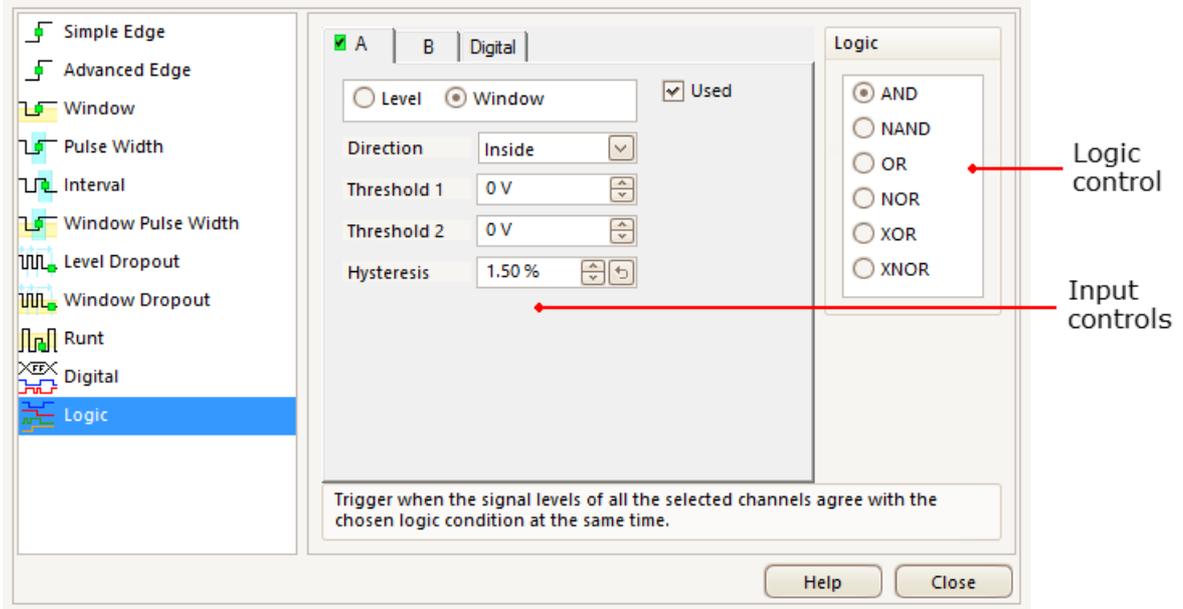
- X** = don't care
- 0** = binary 0
- 1** = binary 1
- R** = rising edge
- F** = falling edge

7.10.2.3 Logic trigger dialog

Location: [Advanced triggering dialog](#) >  **Logic button**

Purpose: sets up triggering on a combination of inputs

Applicability: all devices with more than one active input

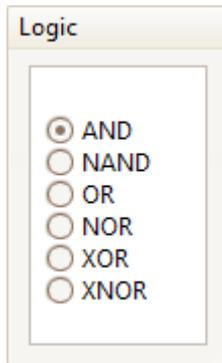


Input controls

There is a set of controls for each active input of the oscilloscope. The selection of inputs depends on the model of oscilloscope in use. The selection of controls (thresholds, hysteresis, window mode and so on) for each input also depends on the hardware capabilities of the oscilloscope.

- Channel A
- Channel B
- Channel C
- Channel D
- EXT input ([if present](#))
- AUX input ([if present](#))
- Digital inputs ([mixed-signal oscilloscopes](#) only). The controls here are the same as those in the [Digital Trigger dialog](#).
- Used Check this box to include the relevant input in the Logic Trigger condition. If the box is not checked, the input will be ignored by the Logic Trigger.

Logic control



Specifies the Boolean operation used to combine the input trigger conditions. Only inputs with the **Used** box checked (see above) are included in the trigger logic.

AND: all of the input trigger conditions must be met

NAND: none of the input trigger conditions must be met

OR: one or more of the input trigger conditions must be met

NOR: none of the input trigger conditions must be met

XOR: an odd number of input trigger conditions must be met

XNOR: an even number of the input trigger conditions must be met

7.11 Zooming and Scrolling toolbar

The **Zooming and Scrolling toolbar** allows you to move around a [scope view](#) or [spectrum view](#). Each button has a keyboard shortcut, as listed below.



 **Ctrl+S** **Normal Selection tool.** Restores the pointer to its normal appearance. You can use this pointer to click buttons, drag [rulers](#) and operate any other controls in the PicoScope window.
or
Esc

 **Ctrl+D** **Hand tool.** Turns the pointer into a hand () that you can use to click and drag the view to pan it vertically and horizontally when you are zoomed in. You can also pan using the scroll bars. Press the **Esc** key to return to the **Normal Selection tool**.

 **Ctrl+M** **Marquee Zoom tool.** This button turns the pointer into a marquee zoom tool: . Use it to draw a box (called a marquee) on the view and PicoScope will magnify that box to fill the view. Scroll bars will appear, which you can drag to pan around in the view, or you can pan by using the **Hand tool** (see above). Zooming in also opens the [Zoom Overview](#) window. Press the **Esc** key to return to the **Normal Selection tool**.

If you point to the time axis, the pointer changes into the horizontal marquee zoom tool () , which restricts zooming to the horizontal axis. This lets you zoom in by an arbitrary amount without disturbing the vertical zoom factor.

Holding down the **Ctrl** key while dragging also restricts zooming to the horizontal axis.

 **Ctrl+I** **Zoom-in tool.** Turns the pointer into a zoom-in tool: . Click on the view with this tool to zoom in to the specified location. Zooming in also opens the [Zoom Overview](#) window.

If you point to the time axis, the pointer changes into the horizontal zoom-in tool () , which restricts zooming to the horizontal axis. This lets you zoom in without disturbing the vertical zoom factor.

Holding down the **Ctrl** key while dragging also restricts zooming to the horizontal axis.

Holding down the **Shift** key while dragging changes the zoom mode to 'zoom out'.

 **Ctrl+O** **Zoom-out tool.** Turns the pointer into a zoom-out tool: . Click on the view with this tool to zoom out around the specified location.

If you point to the time axis, the pointer changes into the horizontal zoom-out tool () , which restricts zooming to the horizontal axis. This lets you zoom out without disturbing the vertical zoom factor.

Holding down the **Ctrl** key while dragging also restricts zooming to the horizontal axis.

Holding down the **Shift** key while dragging changes the zoom mode to 'zoom in'.



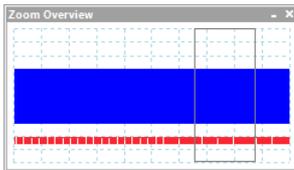
Undo zoom. Returns the view to the previous zoom and pan settings.



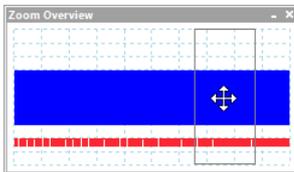
Ctrl+U Zoom to full view. Resets the view to normal size. The view will no longer have scroll bars, and panning will no longer be possible.

7.11.1 Zoom Overview window

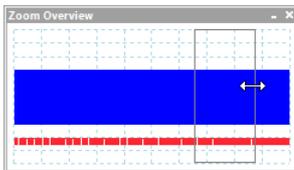
Whenever you zoom in using the [Zooming and Scrolling toolbar](#), the **Zoom Overview** window should appear*:



The **Zoom Overview** shows the full waveforms on all enabled channels. The rectangle indicates the area that is visible in the current view.



You can move around the waveform by dragging the rectangle.



You can also adjust the zoom factor by dragging the edges of the rectangle to resize it.



Minimize button: reduce the **Zoom Overview** window in size without affecting the zoom settings.



Close button: close the **Zoom Overview** window and return the zoom factor to 100%.

*Note: if the **Zoom Overview** does not appear, the feature may have been switched off. Check the **Zoom Overview** option in [Tools](#) > [Preferences](#) > [Options](#).

8 How to...

This chapter explains how to perform some common tasks.

8.1 How to change to a different device

1. Unplug the old [device](#).
2. Cancel the **Check USB cable** dialog.
3. Plug in the new device.
4. PicoScope will detect the new device and begin to use it.

Alternatively, close and reopen the PicoScope software after plugging in the new device.

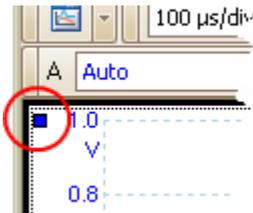
8.2 How to use rulers to measure a signal

Using a single ruler for signal-to-ground measurements

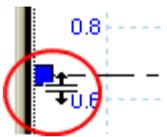
- Look at the [Channels toolbar](#) to find the color code for the [channel](#) you wish to measure:



- Find the ruler handle (the small colored square in the top-left or top-right corner of the [scope view](#) or [spectrum view](#)) of this color:



- Drag the ruler handle downwards. A [signal ruler](#) (horizontal broken line) will appear across the view. Release the ruler handle when the ruler is where you want it.



- Look at the [ruler legend](#) (the small table that appears on the view). It should have a row marked by a small colored square matching the color of your ruler handle. The first column shows the signal level of the ruler.

| 1 | 2 | Δ | - |
|-----------|-----|-----|---|
| ■ 586.0mV | -.- | -.- | |

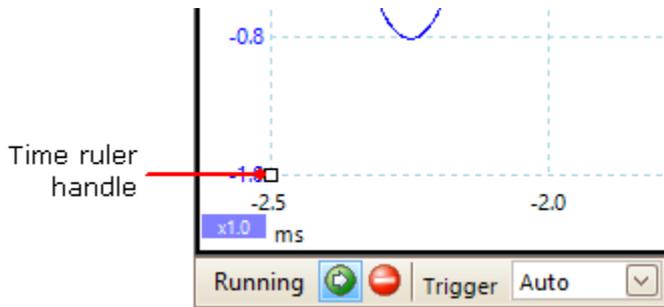
Using two rulers for differential measurements

- Follow the steps above for using a single ruler.
- Drag the second ruler handle of the same color downwards until its ruler is at the signal level to be measured.
- Look at the [ruler legend](#) again. The second column now shows the signal level of the second ruler, and the third column shows the difference between the two rulers.

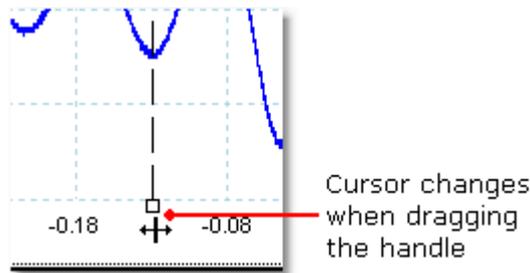
| 1 | 2 | Δ | - |
|-----------|----------|--------|---|
| ■ 586.0mV | -493.0mV | 1.079V | |

8.3 How to measure a time difference

- Find the time ruler handle (the small white square in the bottom left corner of the [scope view](#)).



- Drag the ruler handle to the right. A [time ruler](#) (vertical broken line) will appear on the scope view. Release the ruler handle when the ruler is at the time you wish to use as the reference.



- Drag the second white ruler handle to the right until its ruler is at the time to be measured.
- Look at the [ruler legend](#) (the small table that appears on the scope view). It should have a row marked by a small white square. The first two columns show the times of the two rulers, and the third column shows the time difference.

| | 1 | 2 | Δ | - |
|--------------------------|----------------|---------------|--------------|---|
| <input type="checkbox"/> | -129.0 μ s | -44.0 μ s | 85.0 μ s | |

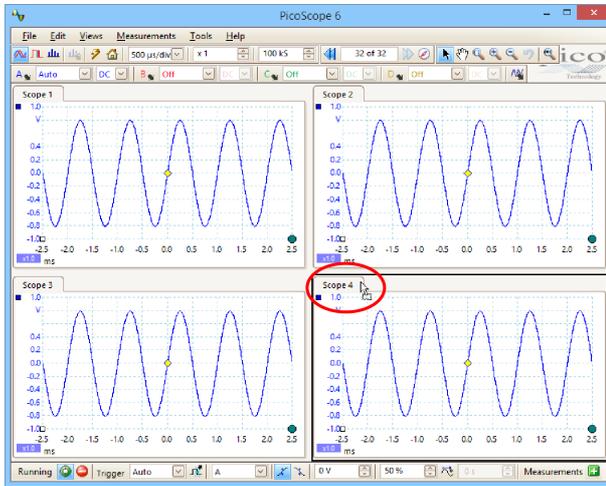
- The [frequency legend](#) shows $1/\Delta$, where Δ is the time difference.

$1/\Delta$ 33.37 Hz, 2002.0 RPM

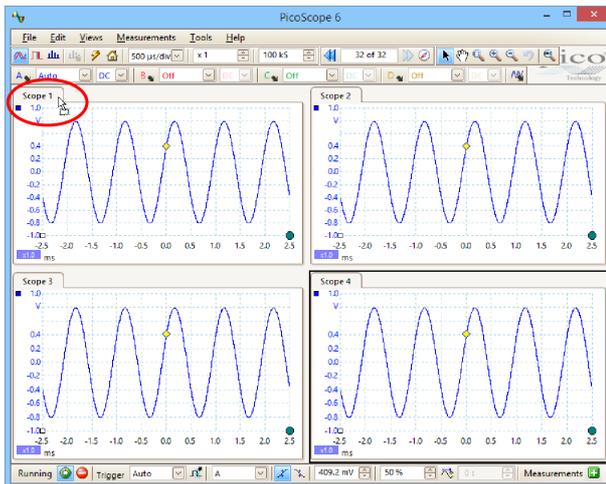
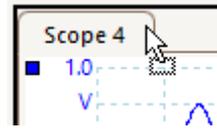
You can use a similar method to measure a frequency difference on a [spectrum view](#).

8.4 How to move a view

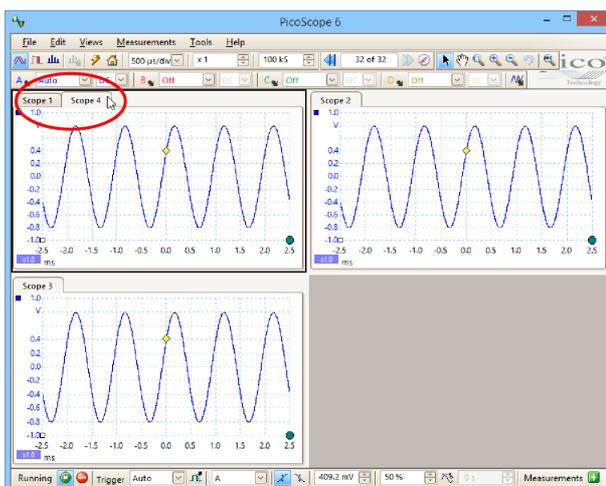
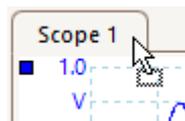
You can easily drag a [view](#) from one [viewport](#) to another. This example shows four view ports, which contain [scope views](#) called **Scope 1** to **Scope 4**. Suppose that you wish to move the **Scope 4** view to the top left view port.



1. Click on the name tab of the Scope 4 view and hold the mouse button down.



2. Drag the mouse pointer to the new location next to the name tab of the Scope 1 view.



3. Release the mouse button, and the view will move to the new location.



8.5 How to scale and offset a signal

PicoScope offers several ways to change the size and position of a signal during or after capture. These methods apply equally to [scope views](#) and [spectrum views](#). They do not change the stored data, only the way in which it is displayed. These options are provided in addition to the [analog offset](#) capability of some scopes (see [Device feature table](#)).

Global zooming and scrolling

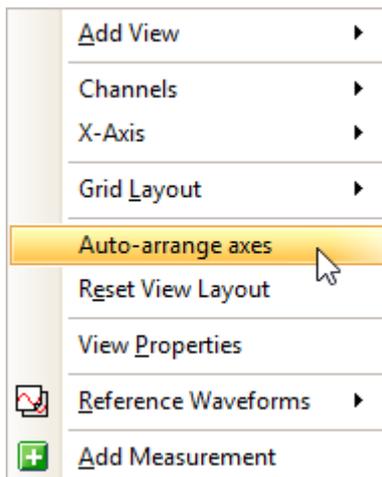
This is usually the quickest way to get a closer look at the fine detail on your signals. The global zooming and scrolling tools move all the signals at once and are found on the [zooming and scrolling toolbar](#).



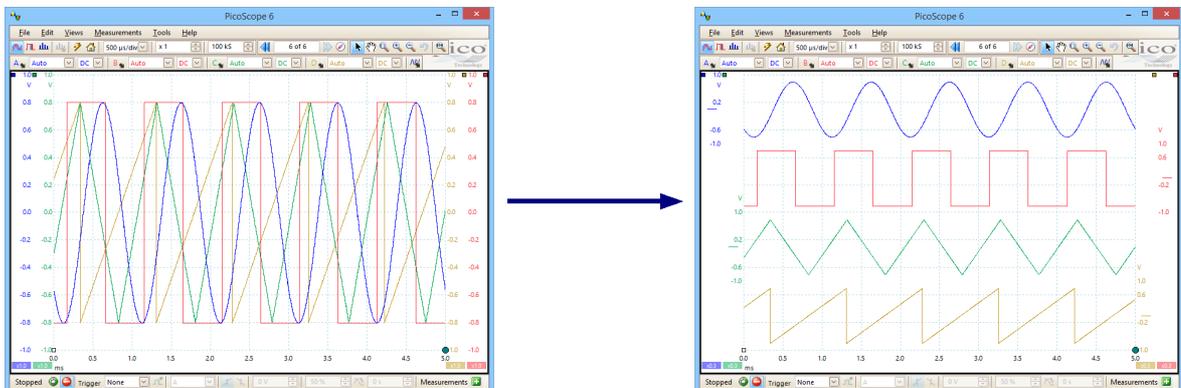
When a view is zoomed in, it has vertical and horizontal scroll bars that let you move the signals around as a group. You can also use the hand tool to scroll around the graph.

Auto-arrange axes

Right-click the scope or spectrum view and select [Auto-arrange axes](#):



PicoScope automatically scales and offsets the channels to make them fit on the view without overlapping. This is the quickest way to tidy up the scope view:

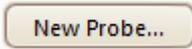


Axis scaling and offset

Use these tools if **Auto-arrange axes** (see above) does not give you the results you want. It allows you to position channels individually on the view (unlike the global zooming and scrolling tools, which are applied to all of the channels at the same time).

Click the scaling button  at the bottom of the axis you wish to modify, and the [axis scaling controls](#) will appear. To adjust the offset without using the axis scaling controls, click on the vertical axis and drag it up or down.

How is this different from scaling my data with a Custom Probe?



You can create a [Custom Probe](#) to apply scaling to the raw data. A Custom Probe may change the scale and position of data on the graph but it has a few important differences from the other scaling methods.

- Custom Probe scaling is a permanent transformation. The scaling is applied when the waveform is captured and cannot be changed afterwards.
- The actual data values themselves are changed, so the graph axes may no longer display the original voltage range of the device.
- Custom Probe scaling can be nonlinear and so may alter the shape of the signal.

Custom Probes are useful when you want to represent the characteristics of a physical probe or transducer that you plug into your scope device. All of the zooming, scrolling, scaling and offset tools still apply to data that has been scaled with a Custom Probe in exactly the same way that they would apply to the raw data.

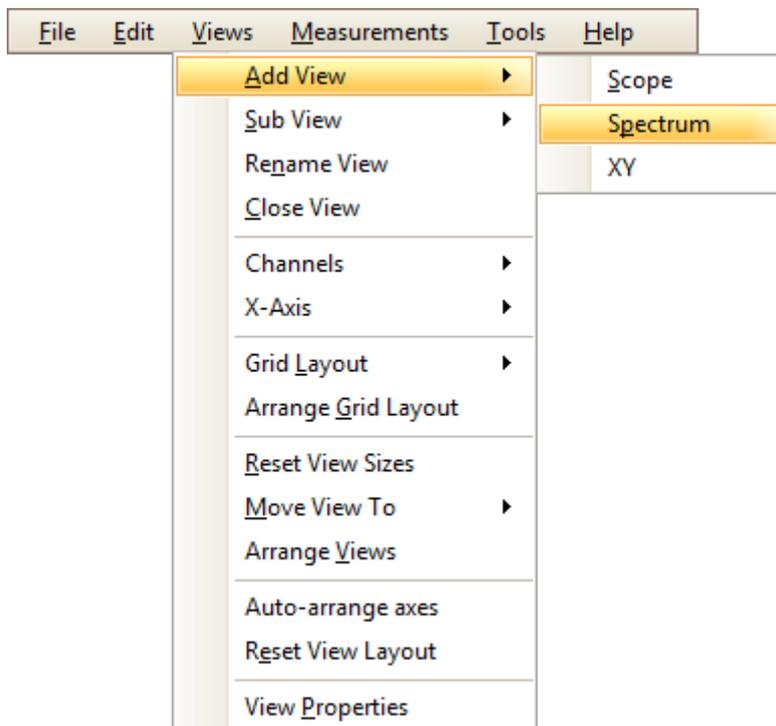
8.6 How to set up the spectrum view

Creating a spectrum view

First, ensure that the [trigger mode](#) is not set to [ETS](#), as it is not possible to open a spectrum view in ETS trigger mode.

There are three ways to open a [spectrum view](#):

- Click the **Spectrum Mode button** in the [Capture Setup toolbar](#). We recommend using this method to get the best spectrum analysis performance from your scope. Once in Spectrum Mode, you can still open a scope view to see your data in the time domain, but PicoScope optimizes the settings for the spectrum view.
- Go to the [Views menu](#) select **Add view**, then select **Spectrum**.



This method opens a spectrum view in the currently selected mode, whether this is Scope Mode or Spectrum Mode. For best results, we recommend that you switch to Spectrum Mode, as described in the method immediately above.

- Right-click on any [view](#), select **Add view**, then select **Spectrum**. The menu is similar to the [Views menu](#) shown above.

Configuring the spectrum view

See [Spectrum Settings dialog](#).

Selecting the source data

PicoScope can produce a [spectrum view](#) based on either live or stored data. If PicoScope is running (the [Start](#) button is pressed in), the spectrum view represents live data. Otherwise, with PicoScope stopped (the [Stop](#) button pressed in), the view represents data stored in the currently selected page of the waveform buffer. When PicoScope is stopped, you can use the [buffer controls](#) to scroll through the buffer and the spectrum view will be recalculated from the waveform currently selected.

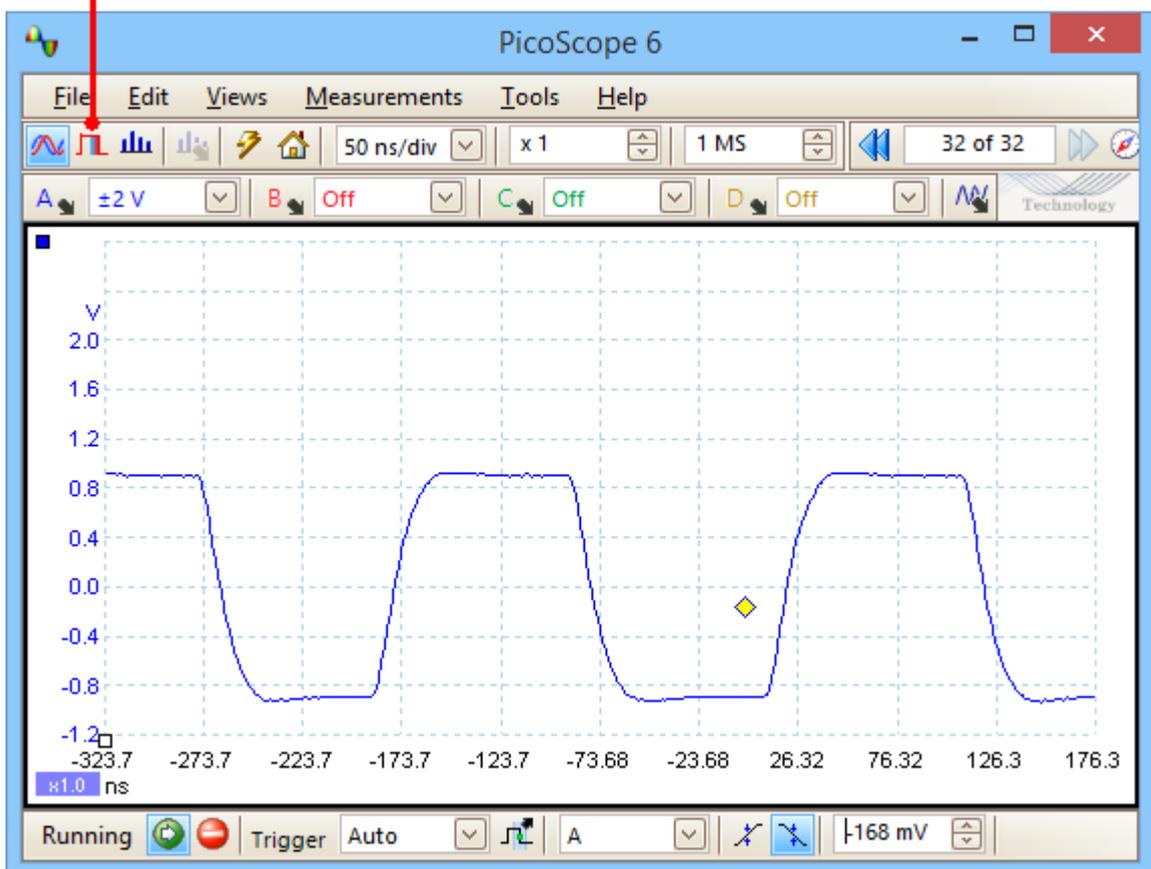
8.7 How to find a glitch using persistence mode

[Persistence mode](#) helps you find rare events hidden in otherwise repetitive waveforms. In normal scope mode, such an event may appear on the display for a fraction of a second, too quickly for you to press the space bar to freeze it on the screen. Persistence mode keeps the event on the display for a predetermined time, allowing you to set up the trigger options to capture it more reliably.

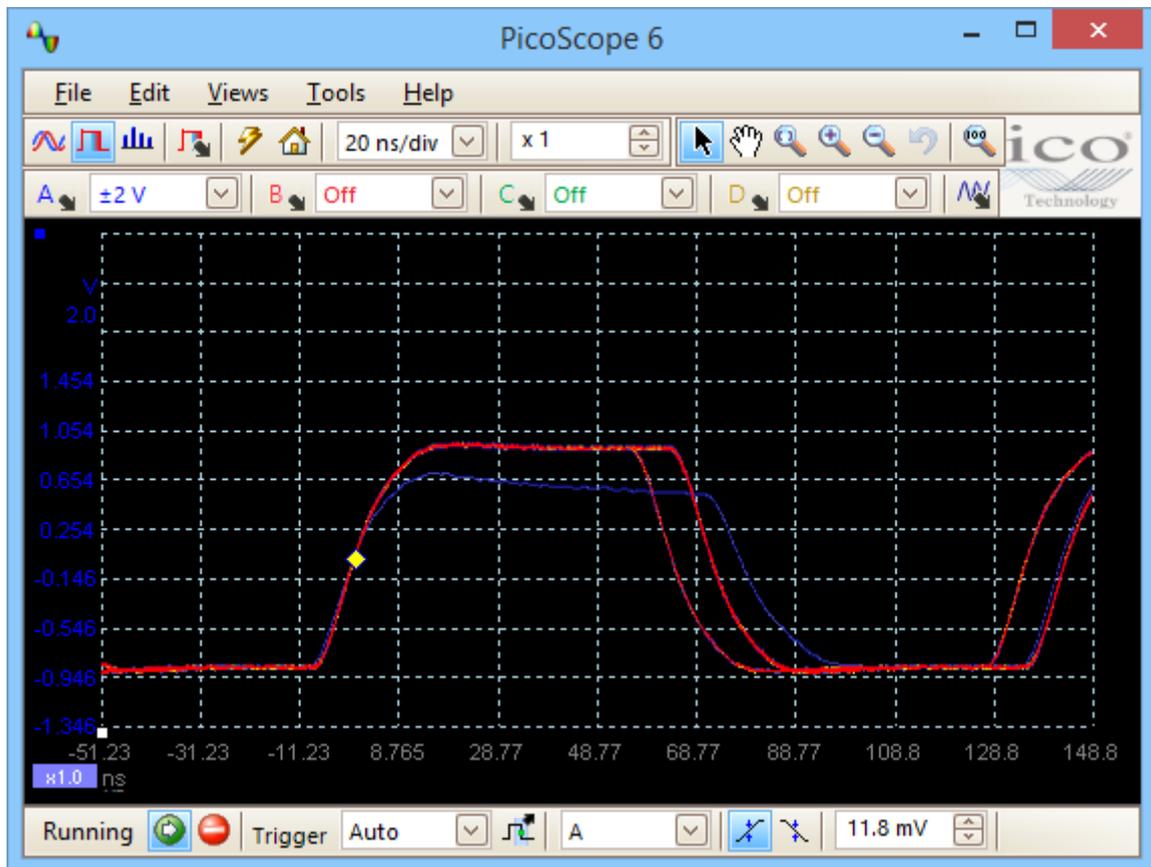
Step-by-step guide

- Set up the scope to trigger on a repetitive waveform like the one below. We suspect that there are occasional glitches but we can see nothing wrong yet, so we shall use persistence mode to investigate. Click the [Persistence Mode button](#) to continue.

Persistence Mode button



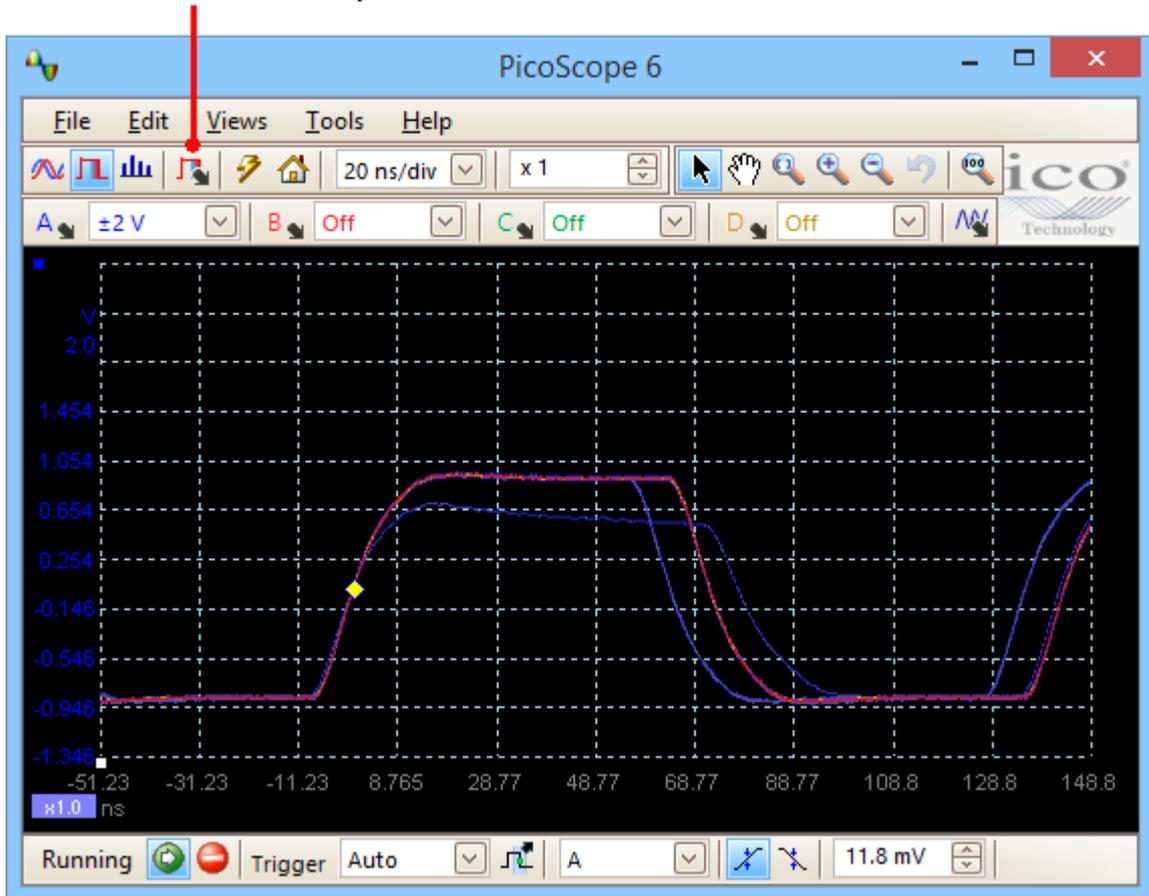
- Our original scope view is replaced by a persistence view, as shown below. Immediately, we can see three pulses with different shapes. At this point we have the **Saturation** control in [Persistence Options](#) turned up to maximum to help us spot the various waveforms easily.



- Now that we have found some glitches, we will turn the **Saturation** control down to minimum. Click the **Persistence Options** button to open the [Persistence Options dialog](#), and then use the slider to adjust the saturation. The display then appears as below.

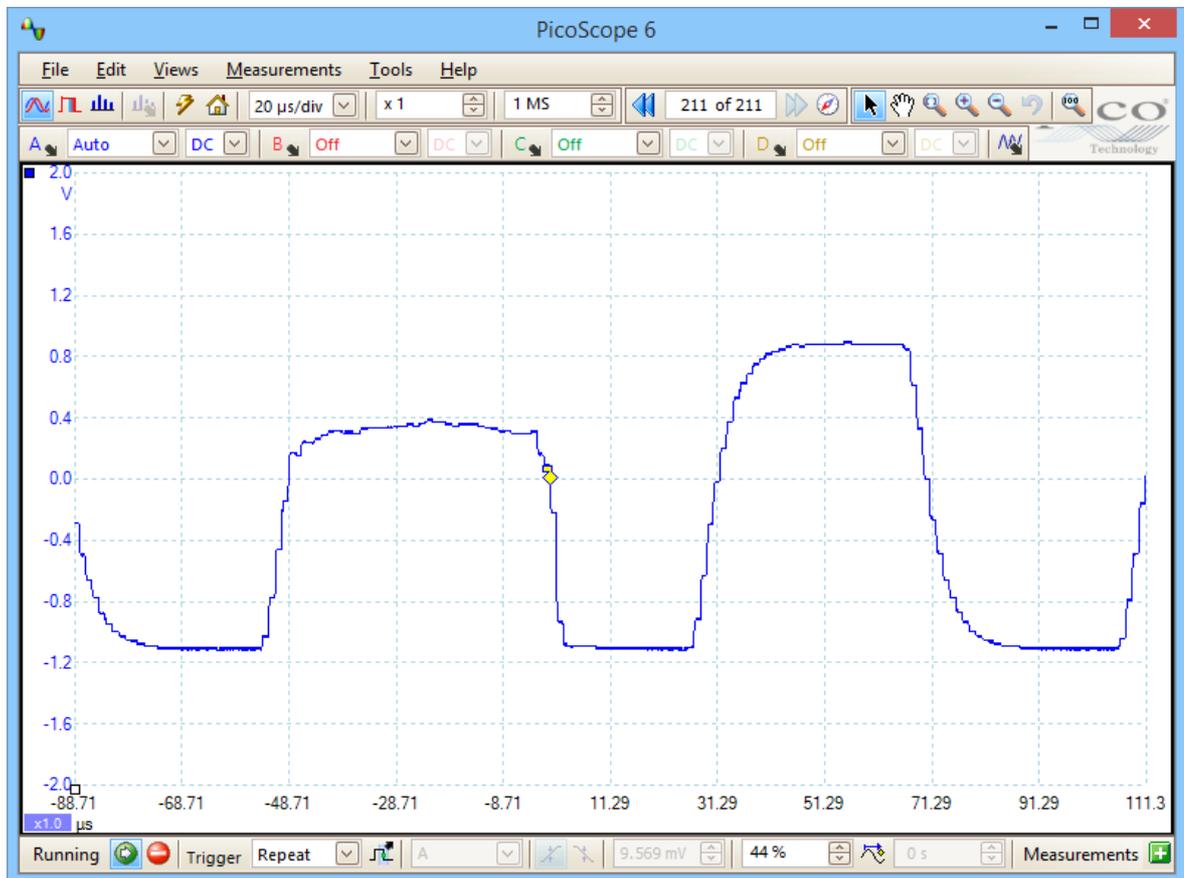
The waveforms are now darker but have a wider range of colors and shades. The most frequently occurring waveform is shown in red, and is the normal shape of the pulse. A second waveform is drawn in light blue to show that it occurs less frequently, and it shows us that there is an occasional jitter of about 10 ns in the pulse width. The third waveform is drawn in dark blue because it occurs less frequently than the other two, and indicates that there is an occasional runt pulse about 300 mV lower in amplitude than normal.

Persistence Options button



- Persistence mode has done its job. We have found our glitches, and now we want to examine them in more detail. The best way to do this is to switch back to normal [scope mode](#), so that we can use the [advanced triggering](#) and [automatic measurement](#) functions built in to PicoScope.

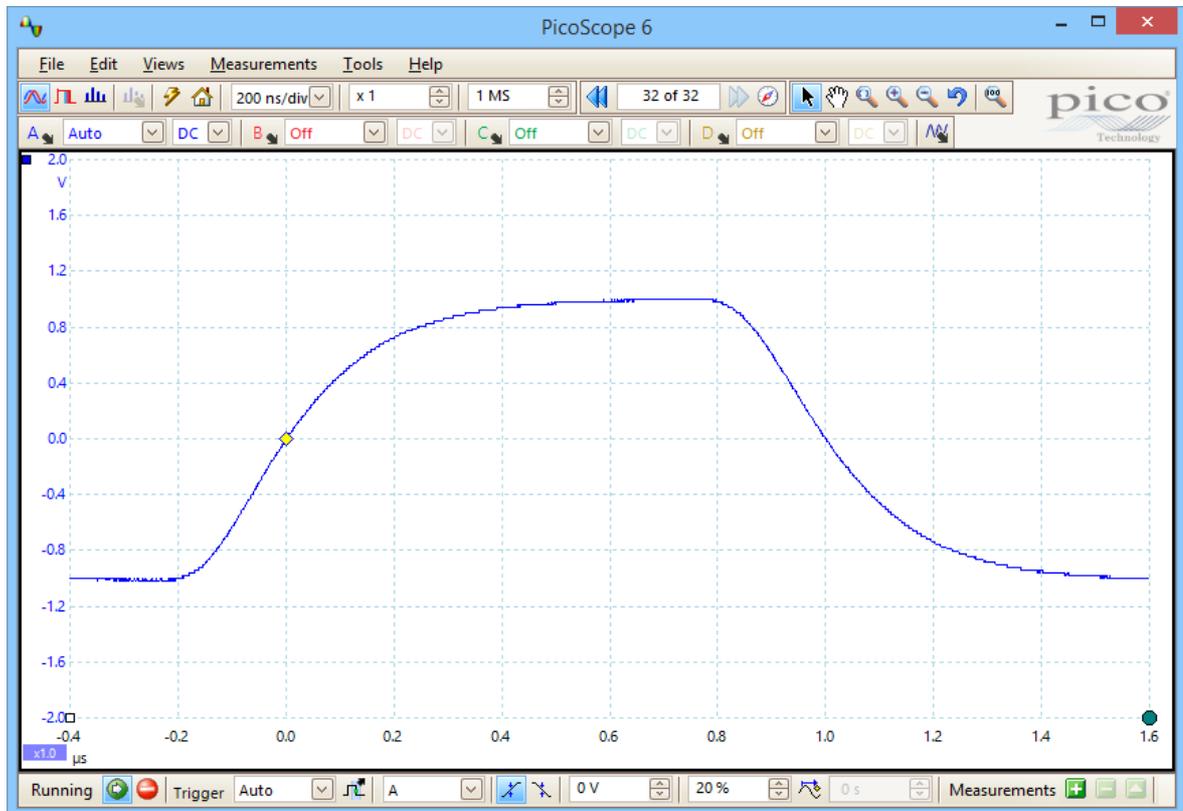
Click the **Scope Mode** button. Set up an advanced pulse-width trigger to look for a pulse wider than 60 ns. PicoScope then finds the runt pulse straight away.



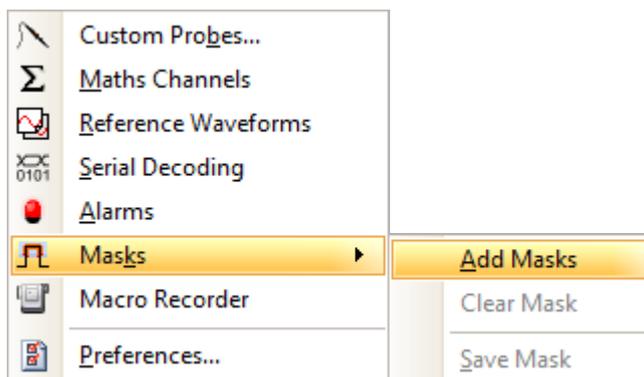
We can now add automatic measurements or drag the rulers into place to analyze the runt pulse in detail.

8.8 How to set up a Mask Limit Test

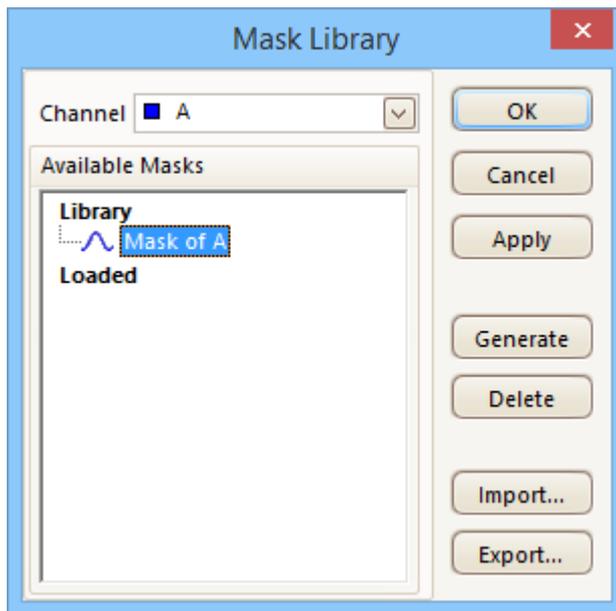
1. Display a stable waveform in a [scope view](#). Adjust the voltage range and timebase so that the feature of interest fills most of the view. In this example, we are viewing a repetitive pulse as might be found on a data bus.



2. Select the **Tools > Masks > Add Masks** command.

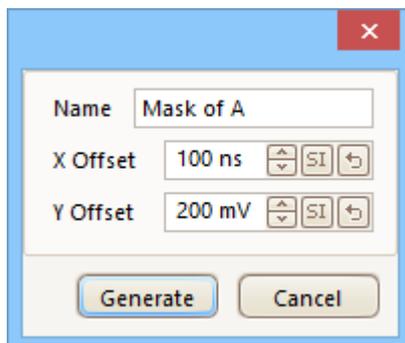


3. You should now be in the [Mask Library dialog](#):

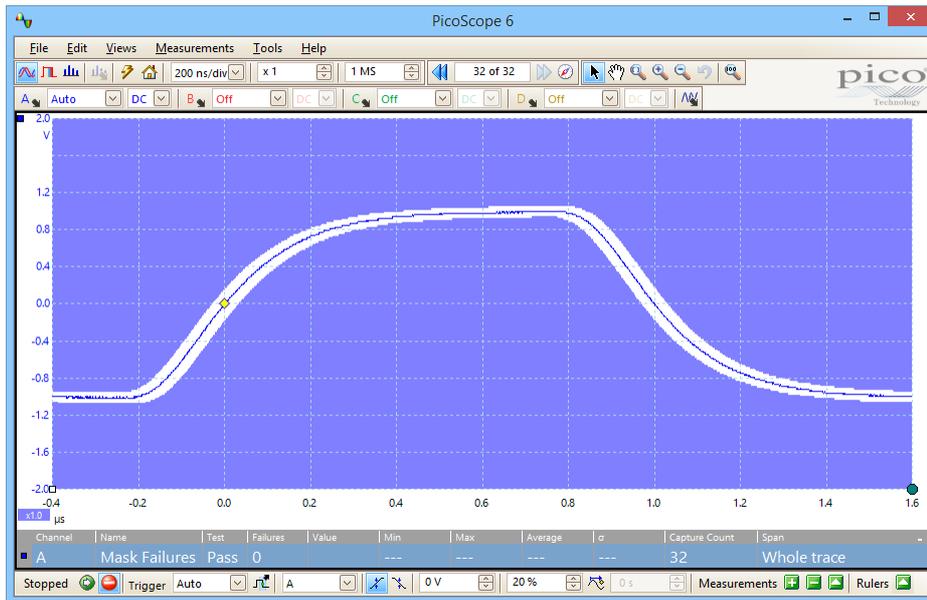


Channel A is selected by default. You can change this if you want to apply the mask to a different channel.

4. Click the **Generate** button to open the [Generate Mask dialog](#):

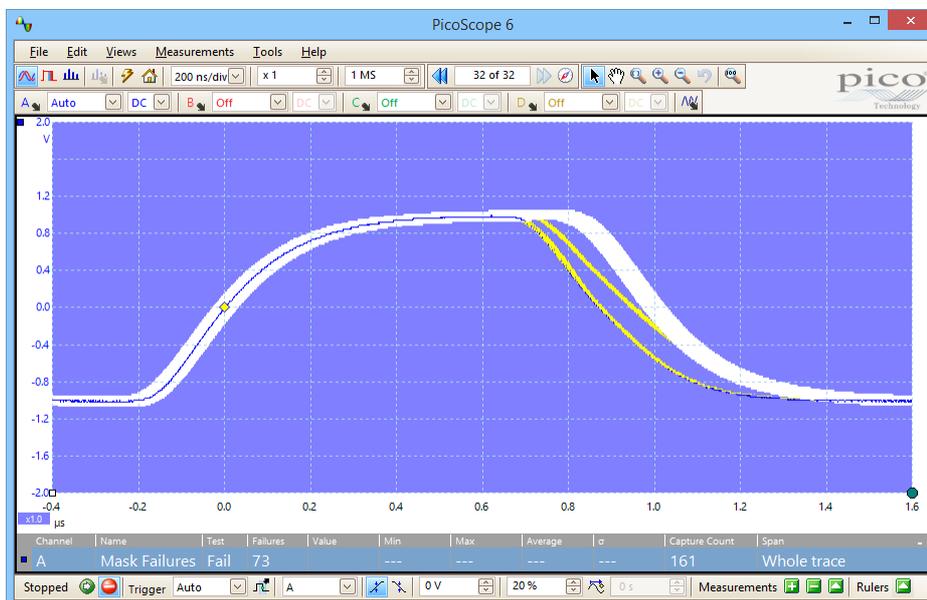


5. For now, accept the default settings and click **Generate**. Then click **OK** in the [Mask Library dialog](#) to return to the scope view:



You now have a mask drawn around the original waveform.

6. PicoScope stops capturing when you enter the [Mask Library dialog](#), so press the space bar to restart. If any captured waveform fails to fit inside the mask, the offending parts are drawn in a contrasting color. The [Measurements table](#) shows the number of failures:



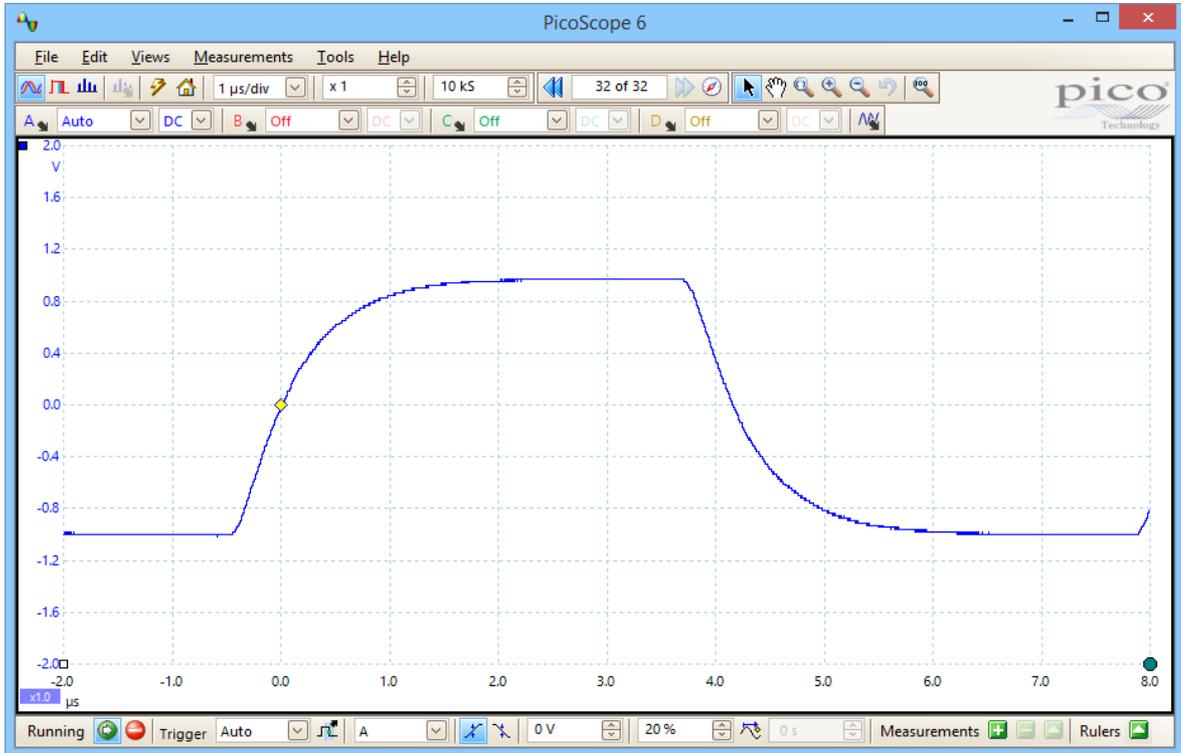
7. You now have a functioning mask limit test. Please read the [Mask Limit Test](#) topic for information on editing, importing and exporting masks. It is also possible to set up a Mask Limit Test on a [spectrum](#) or [XY](#) view.

For further information on this feature, see: [Mask Limit Testing](#).

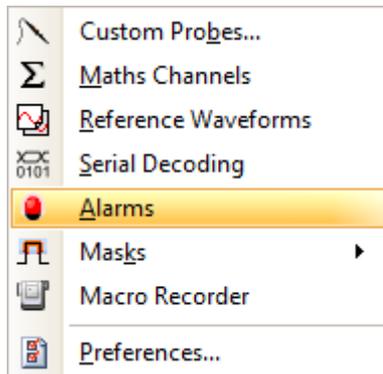
8.9 How to save on trigger

Save-on-trigger is just one of a number of functions that are possible with the [Alarms](#) feature.

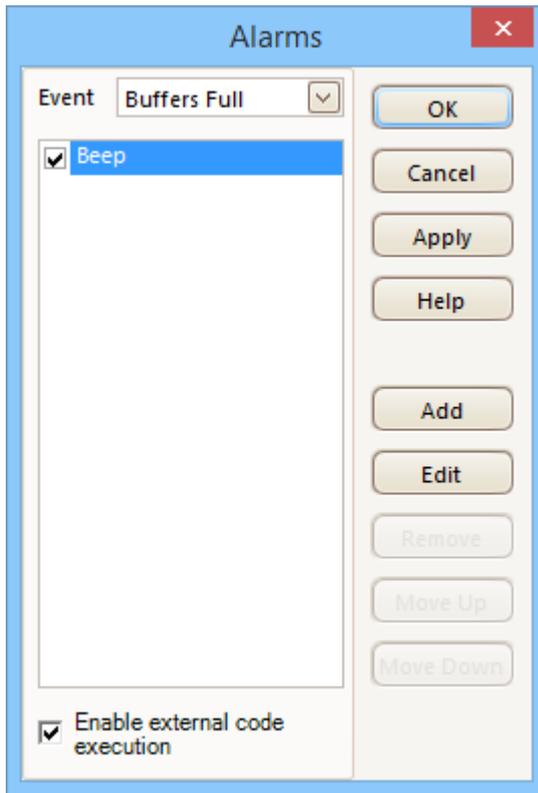
1. Set up PicoScope to display your waveform, and enable triggering:



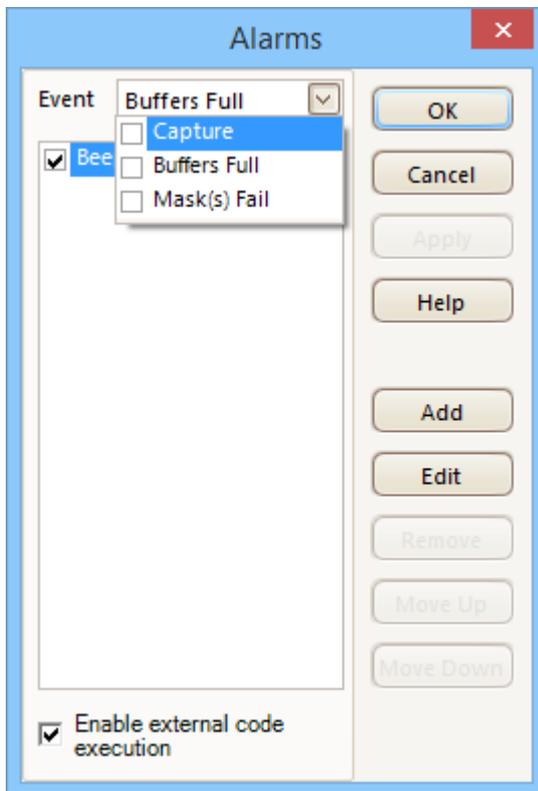
2. Select the **Tools** > **Alarms** command:



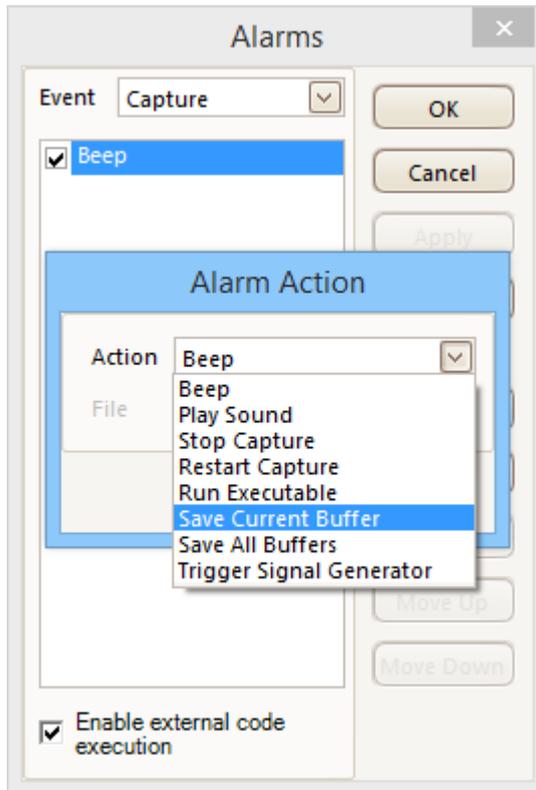
3. You should now be in the **Alarms dialog**:



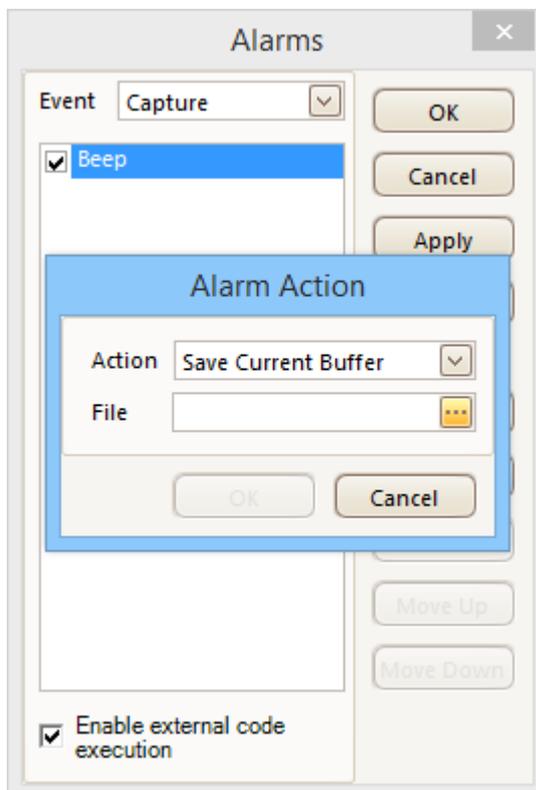
4. Set **Event** to **Capture**:



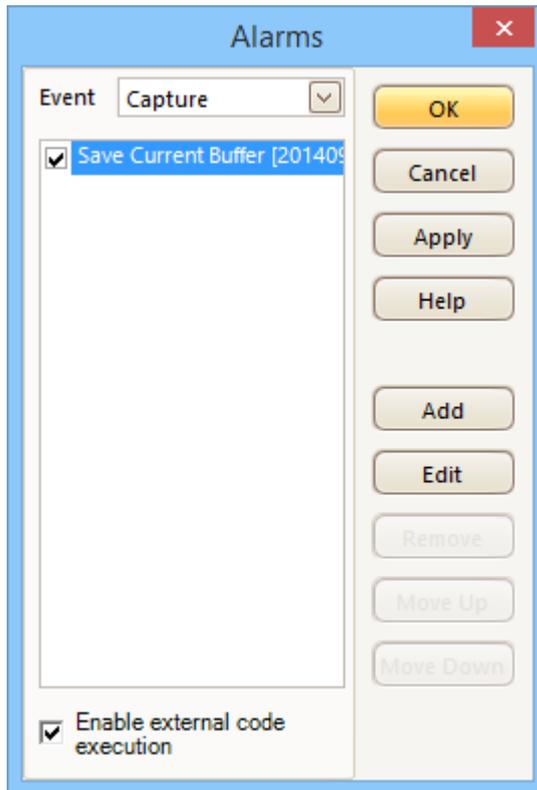
5. Select the first item in the **Actions** list, click **Edit**, and change **Action** to **Save Current Buffer**:



6. Click the  button to the right of the **File** box and enter the name and location of the file to save:



7. Ensure that both the **Save Current Buffer** check box and the **Enable Alarm** check box are set:



8. Click **OK**. PicoScope will now save a file on every trigger event.
9. Switch off the alarm when you have finished using it, to avoid creating unwanted files.

8.10 How to create a link file

A **link file** is used by the [Serial Decoding Table view](#) to convert numerical data to text. This is useful for displaying decoded packets in a human-readable format.

1. In the **Tools > Serial Decoding** menu, enable serial decoding and select the required serial protocol. In this example we shall use I²C. Ensure that the **Table** view is enabled.
2. PicoScope will show the decoded data in numeric format:

| No. | Packet | Address | Read/Write | Data bytes | Acknowledge | Start Time | End Time |
|-----|------------|---------|------------|------------|-------------|------------|-----------|
| 1 | Start | - | - | - | - | -20.09 µs | -20.09 µs |
| 2 | Address(7) | 0D | - | - | 0 | 5.558 ns | 20.71 µs |
| 3 | Data | - | - | 02 | 0 | 54.31 µs | 75.01 µs |
| 4 | Data | - | - | 28 | 0 | 106 µs | 126.6 µs |
| 5 | Data | - | - | A3 | 0 | 157.6 µs | 178.3 µs |

2. To create a link file, begin by creating a blank template. Click the **+** next to the **Link** label in the **Serial Data Table** toolbar:



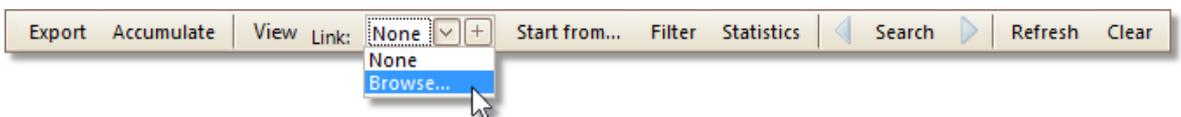
A **Save As** dialog will appear. Choose the **Save As Type** (.xls for Microsoft Excel format, or .csv for comma-separated text format) and then save the link file to a convenient place such as your Windows desktop.

3. Open the link file with an appropriate text or spreadsheet editor, and add number/string pairs appropriate to the target system:

| Address | Address Description | Read/Write | Read/Write Description | Data bytes | Data bytes Description | Acknowledge | Acknowledge Description |
|---------|---------------------|------------|------------------------|------------|------------------------|-------------|-------------------------|
| 0D | ModeReg | | | 02 | Play | | |
| | | | | 28 | Stop | | |
| | | | | A3 | Pause | | |

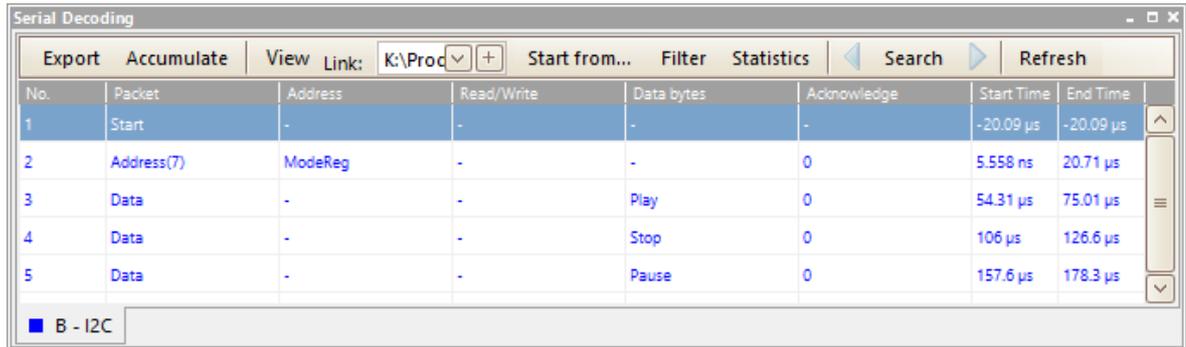
In this example we have specified that the string 'ModeReg' will appear in place of the value '0D' in the **Address** field, and so on. The numerical values are treated literally, so if the **View > Display Format** control in the data window is set to **Hex**, the link file must contain the string '0D' to match the hexadecimal value 0D. If the **View > Display Format** control is changed to **Decimal**, for example, a new link file containing decimal values will be required.

4. Save the link file.
5. Click **Browse** next to the **Link** label in the **Serial Data Table** toolbar:



6. An **Open** dialog will appear. Select the link file and click **OK**.

- PicoScope will now replace the numeric data with the text strings that you have defined:



The screenshot shows the 'Serial Decoding' window in PicoScope. The window title is 'Serial Decoding'. At the top, there are buttons for 'Export', 'Accumulate', 'View', and 'Link: K:\Proc'. Below these are 'Start from...', 'Filter', 'Statistics', 'Search', and 'Refresh'. The main area is a table with the following columns: 'No.', 'Packet', 'Address', 'Read/Write', 'Data bytes', 'Acknowledge', 'Start Time', and 'End Time'. The table contains five rows of data. Below the table, there is a label 'B - I2C' with a blue square icon.

| No. | Packet | Address | Read/Write | Data bytes | Acknowledge | Start Time | End Time |
|-----|------------|---------|------------|------------|-------------|----------------|----------------|
| 1 | Start | - | - | - | - | -20.09 μ s | -20.09 μ s |
| 2 | Address(7) | ModeReg | - | - | 0 | 5.558 ns | 20.71 μ s |
| 3 | Data | - | - | Play | 0 | 54.31 μ s | 75.01 μ s |
| 4 | Data | - | - | Stop | 0 | 106 μ s | 126.6 μ s |
| 5 | Data | - | - | Pause | 0 | 157.6 μ s | 178.3 μ s |

■ B - I2C

9 Reference

This is where you can find detailed information on the operation of PicoScope.

9.1 Measurement types

The [Edit Measurement dialog](#) presents a selection of measurements that PicoScope can calculate for the selected view.

9.1.1 Scope measurements

AC RMS. The root mean square (RMS) value of the waveform *minus* the **DC Average**. It is equivalent to a *ripple* measurement.

Cycle Time. PicoScope will attempt to find a repeated pattern in the waveform and measure the duration of one cycle.

DC Average. The mean value of the waveform.

Duty Cycle. The amount of time that a signal spends above its mean value, expressed as a percentage of the signal period. A duty cycle of 50% means that the high time is equal to the low time.

Falling Rate. The rate at which the signal level falls, in signal units per second. Click the **Advanced** button in the **Add Measurement** or **Edit Measurement** dialog to specify the signal level thresholds for the measurement.

Frequency. The number of cycles of the waveform per second.

Fall Time. The time the signal takes to fall from the upper threshold to the lower threshold. Click the **Advanced** button in the **Add Measurement** or **Edit Measurement** dialog to specify the signal level thresholds for the measurement.

High Pulse Width. The amount of time that the signal spends above its mean value.

Low Pulse Width. The amount of time that the signal spends below its mean value.

Maximum. The highest level reached by the signal.

Minimum. The lowest level reached by the signal.

Peak To Peak. The difference between **maximum** and **minimum**.

Rise Time. The time the signal takes to rise from the lower threshold to the upper threshold. Click the **Advanced** button in the **Add Measurement** or **Edit Measurement** dialog to specify the signal level thresholds for the measurement.

Rising Rate. The rate at which the signal level rises, in signal units per second. Click the **Advanced** button in the **Add Measurement** or **Edit Measurement** dialog to specify the signal level thresholds for the measurement.

True RMS. The root mean square (RMS) value of the waveform, including the DC component.

Mask Failures. A special measurement that counts the number of failed waveforms in [Mask Limit Testing](#) mode. This measurement is added to the table automatically when you use Mask Limit Testing, so there is usually no need to select it manually.

9.1.2 Spectrum measurements

To add a **spectrum measurement**, open a [spectrum view](#) and then click the [Add Measurement](#) button. You can use these measurements in either [scope mode](#) or [spectrum mode](#).

Frequency at peak. The frequency at which the peak signal value appears.

Amplitude at peak. The amplitude of the peak signal value.

Average amplitude at peak. The amplitude of the peak signal value averaged over a number of captures.

Total power. The power of the whole signal captured in the spectrum view, calculated by adding the powers in all of the spectrum bins.

Total Harmonic Distortion (THD). The ratio of the sum of harmonic powers to the power at the fundamental frequency.

$$THD = 20 \log_{10} \left(\frac{\sqrt{V_2^2 + V_3^2 + V_4^2 + V_5^2 + V_6^2 + V_7^2}}{V_1} \right)$$

Total Harmonic Distortion plus Noise (THD+N). The ratio of the harmonic power plus noise to the fundamental power. THD+N values are always greater than the THD values for the same signal.

$$THD + N = 20 \log_{10} \left(\frac{\sqrt{\text{sum of squares of RMS values excluding datum}}}{\text{RMS value of datum}} \right)$$

Spurious-free Dynamic Range (SFDR). This is the ratio of the amplitude of the specified point (normally the peak frequency component) and the frequency component with the second largest amplitude (call it *SFDR frequency*). The component at the SFDR frequency is not necessarily a harmonic of the fundamental frequency component. For example, it might be a strong, independent noise signal.

Signal+Noise+Distortion to Signal+Noise Ratio (SINAD). The ratio, in decibels, of the signal-plus-noise-plus-distortion to noise-plus-distortion.

$$SINAD = 20 \log_{10} \left(\frac{\text{RMS value of datum}}{\sqrt{\text{sum of squares of all RMS components except datum}}} \right)$$

Signal to Noise Ratio (SNR). The ratio, in decibels, of the mean signal power to the mean noise power. Hanning or Blackman windows are recommended because of their low noise.

$$SNR = 20 \log_{10} \left(\frac{\text{RMS value of datum}}{\sqrt{\text{sum of squares of all values excluding datum and harmonics}}} \right)$$

Intermodulation Distortion (IMD). A measure of the distortion caused by the nonlinear mixing of two tones. When multiple signals are injected into a device, modulation or nonlinear mixing of these two signals can occur. For input signals at frequencies f_1 and f_2 , the two second-order distortion signals will be found at frequencies: $f_3 = (f_1 + f_2)$ and $f_4 = (f_1 - f_2)$.

IMD is expressed as the dB ratio of the RMS sum of the distortion terms to the RMS sum of the two input tones. IMD can be measured for distortion terms of any order, but the second-order terms are most commonly used. In the second-order case, the intermodulation distortion is given by:

$$IMD = 20 \log_{10} \sqrt{\frac{F_3^2 + F_4^2}{F_1^2 + F_2^2}}$$

where

F_3 and F_4 are the amplitudes of the two second-order distortion terms (at frequencies f_3 and f_4 defined above)

and

F_1 and F_2 are the amplitudes of the input tones (at frequencies f_1 and f_2 , as marked by the frequency rulers in the spectrum window).

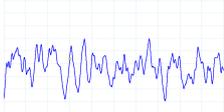
For reference, the third-order terms are at frequencies $(2F_1 + F_2)$, $(2F_1 - F_2)$, $(F_1 + 2F_2)$ and $(F_1 - 2F_2)$.

Note: Hanning or Blackman windows are recommended because of their low noise. An FFT size of 4096 or greater is recommended in order to provide adequate spectral resolution for the IMD measurements.

Mask Failures. See [Mask Limit Testing](#).

9.2 Signal generator waveform types

The list of waveform types available in the [Signal Generator dialog](#) varies according to the type of oscilloscope connected. The full list is as follows:

| | | |
|------------|---|--|
| Sine |  | Sinusoid |
| Square |  | Square wave |
| Triangle |  | Symmetrical triangle wave |
| RampUp |  | Rising sawtooth |
| RampDown |  | Falling sawtooth |
| Sinc |  | $\sin(x)/x$, truncated on the x axis |
| Gaussian |  | The 'bell curve' of the normal distribution, truncated on the x axis |
| HalfSine |  | A rectified sinusoid |
| WhiteNoise |  | Random samples at the maximum update rate of the AWG |
| PRBS |  | Pseudo-random binary sequence - a random sequence of bits with adjustable bit rate |
| DCVoltage |  | Constant voltage, adjustable using the Offset control |
| Arbitrary |  | Any waveform created by the arbitrary waveform editor |

9.3 Spectrum window functions

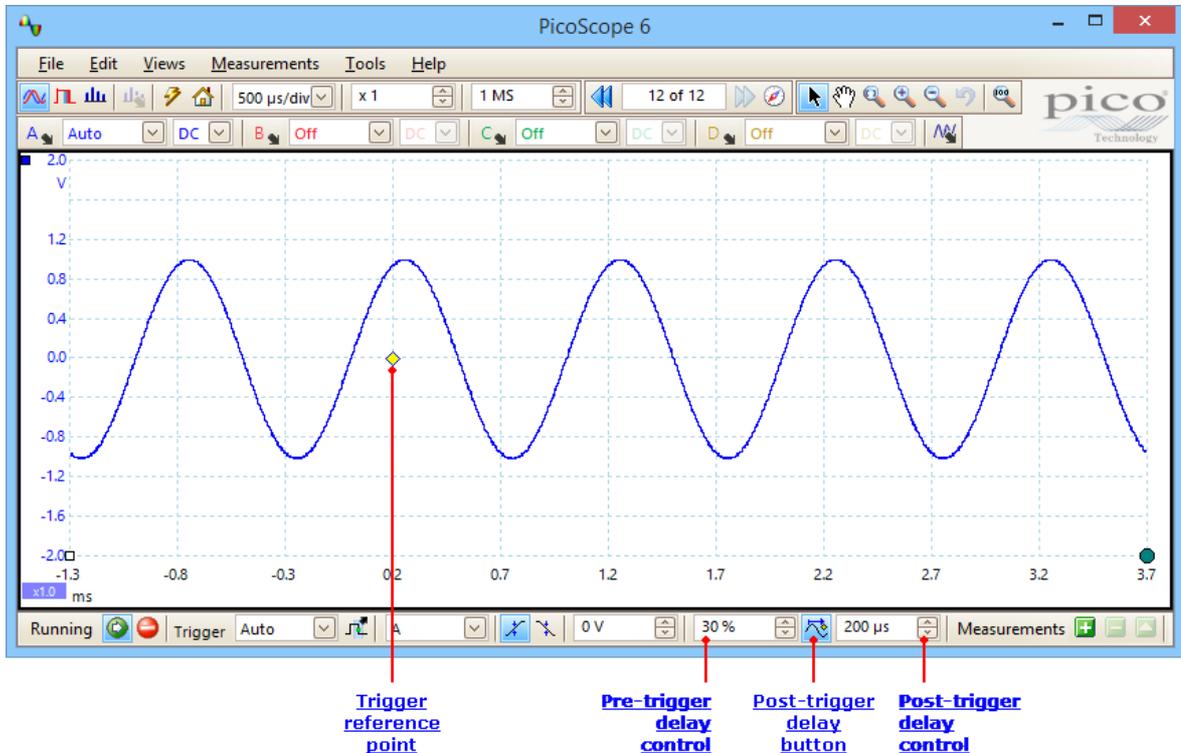
To create a [spectrum view](#), PicoScope captures a block of sampled data over a finite time interval and then uses a Fast Fourier Transform to compute its spectrum. The algorithm assumes a signal level of zero at all times outside the captured time interval. Typically, this assumption causes sharp transitions to zero at either end of the data, and these transitions have an effect on the computed spectrum, creating unwanted artefacts such as ripple and gain errors. To reduce these artefacts, the signal can be faded in and out at the start and end of the block. There are several commonly used "window functions" that can be convolved with the data to effect this fading, and which are chosen according to the type of signal and the purpose of the measurement.

The **Window Functions control** in the [Spectrum Options dialog](#) lets you select one of the standard window functions for spectrum analysis. The following table shows some of the figures of merit used to compare the functions.

| Window | Main peak width (bins @ -3 dB) | Highest side lobe (dB) | Side lobe roll-off (dB/octave) | Notes |
|------------------------|--------------------------------|------------------------|--------------------------------|--|
| Blackman | 1.68 | -58 | 18 | often used for audio work |
| Gaussian | 1.33 to 1.79 | -42 to -69 | 6 | gives minimal time and frequency errors |
| Triangular | 1.28 | -27 | 12 | also called Bartlett window |
| Hamming | 1.30 | -41.9 | 6 | also called raised sine-squared; used in speech analysis |
| Hann | 1.20 to 1.86 | -23 to -47 | 12 to 30 | also called sine-squared; used for audio & vibration |
| Blackman-Harris | 1.90 | -92 | 6 | general-purpose |
| Flat-top | 2.94 | -44 | 6 | negligible pass-band ripple; used mainly for calibration |
| Rectangular | 0.89 | -13.2 | 6 | no fading; maximal sharpness; used for short transients |

9.4 Trigger timing (part 1)

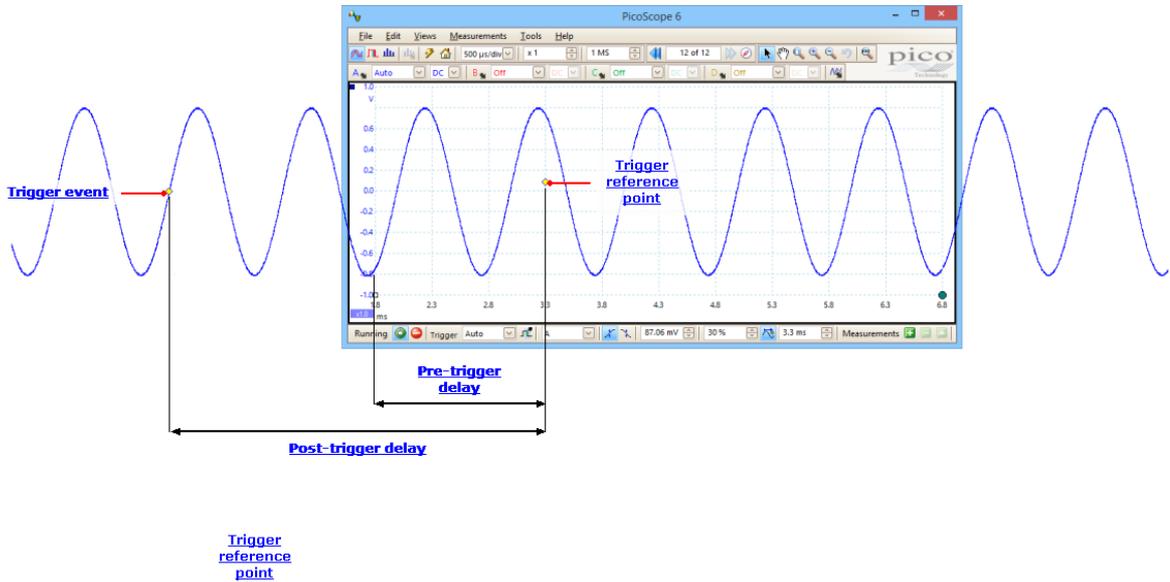
The **pre-trigger time control** and **post-trigger delay control** functions are described individually under [Triggering toolbar](#), but the interaction between the two controls is also important to understand. Here is a screen shot of a [scope view](#) with post-trigger delay enabled:



- Note 1. The trigger reference point (◆) does not lie on the waveform. This is because the post-trigger delay is set to 200 μs, which means that the trigger occurred 200 μs before the reference point, somewhere off the left-hand edge of the [scope view](#). The time axis is aligned so that the trigger reference point is at 200 μs.
- Note 2. The pre-trigger delay is set to 30%, which makes the trigger reference point appear 30% of the way across the scope view from the left-hand edge.
- Note 3. PicoScope limits the trigger-to-reference-point delay to a multiple of the total capture time. Once you have reached this limit, the program will not let you increase the pre-trigger delay, and if you increase the post-trigger delay, PicoScope will reduce the pre-trigger delay to stop the total exceeding the limit. The multiple is typically 100 in most trigger modes, and 1 in [ETS](#) mode.

9.5 Trigger timing (part 2)

[Trigger timing \(part 1\)](#) introduced the concepts of [pre-trigger delay](#) and the [post-trigger delay](#). This diagram below shows how they are related.



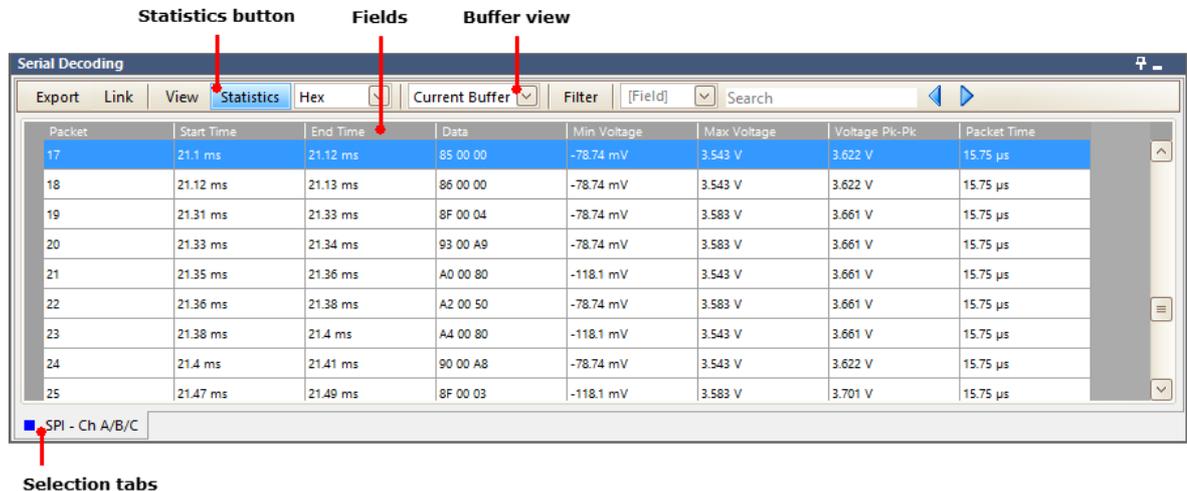
The **pre-trigger delay** positions the [scope view](#) in relation to the trigger reference point so that you can choose how much of the waveform should be before the reference point, and how much after it.

The **post-trigger delay** is like the delayed trigger of a conventional oscilloscope. PicoScope waits for this time after the trigger event before drawing the trigger reference point. Scope devices have a limit to the number of sampling intervals that can elapse between the trigger event and the end of the capture, so the software may adjust the pre-trigger delay to keep within this limit.

Tip: If you have set up a post-trigger delay, you can click the post-trigger delay button while the scope is running whenever you want to switch between viewing the trigger event and the trigger reference point.

9.6 Serial protocols

The [serial decoding](#) feature of PicoScope understands a range of serial protocols. Each one will show different sets of data, as detailed in the following sections. To see all of these data fields, make sure **Table** is selected in the [Protocol settings](#). For more information on any of the protocols, visit the [PicoScope A to Z](#).



There are some fields which every protocol will display.

Packet Serial number of the frame (decimal).

If **All Buffers** is selected, this is counted from the start of the selected waveform.

If **Current Buffer** is selected, this is counted from the start of the first waveform in the current waveform buffer.

Start Time Value of the PicoScope timebase at the start of the frame.

End Time Value of the PicoScope timebase at the end of the frame.

If the **Statistics button** is selected, a further four columns will be displayed.

Min Voltage Minimum voltage

Max Voltage Maximum voltage

Voltage Pk-Pk Voltage range of the signal, from **Max Voltage** to **Min Voltage**.

Packet Time The duration of the packet, from **Start Time** to **End Time**.

9.6.1 1-Wire protocol

1-Wire is a serial protocol originally developed by the Dallas Semiconductor Corporation. It is designed to provide low-speed data rates, signaling, and power.

| | |
|-----------------------|--|
| Reset Pulse | Duration of the reset pulse generated by the master. |
| Presence Pulse | Duration of the presence pulse generated by the slave. |
| Command | The numerical value of the command. |
| Command Code | The name of the command code. |
| Data | The data content of the packet. |

9.6.2 ARINC 429 protocol

ARINC 429, also known as the Mark 33 DITS specification, was developed to provide interchangeability and interoperability of line replaceable units (LRUs) in commercial aircraft. ARINC 429 defines the physical and electrical interfaces of a two-wire data bus and the data protocol to support an aircraft's avionics local area network.

For more information on ARINC 429, visit the [PicoScope A to Z](#).

| | |
|--------------------------|---|
| BNR Data | Binary Data. Formatted according to the SSM bits. |
| BNR STATUS MATRIX | The meaning of the SSM field decoded according to the Label. |
| Label | An 8-bit value indicating the data type. |
| SDI | Source/Destination Identifier. |
| Data | The data content of the word. |
| SSM | Sign/Status Matrix. The meaning of this 2-bit field depends on the Label. |
| Parity Bit | The error-correction bit at the end of the word. |
| Parity Valid | Parity result calculated by PicoScope. |
| Valid | Confirms whether the packet is valid. |

9.6.3 CAN protocol

CAN (Controller Area Network) is a serial protocol used in automotive and industrial machinery to allow microcontrollers to communicate with each other. The standard was originally developed in 1983 by Robert Bosch GmbH. It typically uses differential signaling (with signals named CAN H and CAN L) to increase noise immunity.

| | |
|------------------|---|
| ID | Identifier (hexadecimal). This should be unique for a specified data type. Base frames have an 11-bit identifier, and extended frames have a 29-bit identifier. |
| RTR | Remote transmission request. |
| IDE | ID Extension bit. 0 for standard format packet, 1 for extended format. |
| DLC | Data length code. Indicates the number of bytes of data. |
| Data | The data content of the packet. |
| CRC | A cyclic redundancy check of the data. |
| ACK | Node asserts this bit to acknowledge receipt. |
| CRC Valid | The result of PicoScope's CRC. |

9.6.4 DCC protocol

DCC (Digital Command Control) is a serial protocol used for the control of digital model railway systems.

| | |
|--------------------------|---|
| Address | Shown for address packets. |
| Instructions | Commands sent to multi-function digital decoders. The First three bits determine an instruction type, and the rest describe the actions within this type. |
| Preamble | String of ones indicating start of packet. |
| Error Detection | Error correction byte. |
| Valid | The results of the error check. |
| Address Partition | Assigned based on the type of decoder. Specified in the DCC standard. |
| Instruction Type | The first three bits of the instruction bytes determining the instruction type, as specified in the DCC standard. |
| Data | The rest of the instruction bits determining a specific action for a corresponding instruction type (given in Instruction Type Field). |

9.6.5 DMX512 protocol

DMX512 (Digital Multiplex) is a serial protocol designed for the control of stage lighting and effects.

| | |
|-----------------------------|---|
| Slot Count | Number of channels. |
| Slot 0 Value | Also known as the <i>start code</i> . |
| Slot Range | Numbers of the first and last slots in this packet. |
| Slot Values | An 8-bit value for each slot. |
| Active Channel Count | The number of channels which are above 0. |

9.6.6 Ethernet 10BASE-T protocol

Ethernet 10BASE-T is a serial protocol supporting Ethernet speeds of up to 10 Mbit/s.

| | |
|------------------------------|---|
| Preamble | Ethernet preamble (PRE). |
| Start Frame Delimiter | A fixed bit pattern of 10101011. |
| Destination MAC | Destination address. |
| Source MAC | Source address. |
| Type | Length/Type. If ≤ 1500 , the number of LLC bytes. If > 1536 , the packet type. |
| Data | The data content of the packet. |
| Frame Check Sequence | CRC value used for error detection. |
| Ethertype | Meaning of the Length/Type field if it represents a Type. |
| CRC Pass | Result of the cyclic redundancy check. |

9.6.7 Fast Ethernet 100BASE-TX protocol

Ethernet 100BASE-TX is a serial protocol supporting Ethernet speeds of up to 100 Mbit/s.

| | |
|------------------------------|---|
| Preamble | Ethernet preamble (PRE). |
| Start Frame Delimiter | A fixed bit pattern of 10101011. |
| Destination MAC | Destination address. |
| Source MAC | Source address. |
| Type | Length/Type. If ≤ 1500 , the number of LLC bytes. If > 1536 , the packet type. |
| Data | The data content of the packet. |
| Frame Check Sequence | CRC value used for error detection. |
| Ethertype | Meaning of the Length/Type field if it represents a Type. |
| CRC Pass | Result of the cyclic redundancy check. |

9.6.8 FlexRay protocol

FlexRay is an automotive network communications protocol for high-speed data. It is designed to be fault-tolerant.

| | |
|-----------------------------------|--|
| Header | Header segment. |
| Payload | Payload segment. |
| Trailer | Trailer segment. |
| Reserved Bit | Undefined. |
| Payload Preamble Indicator | Indicates the presence of a network management vector or message ID. |
| Null Frame Indicator | Indicates that the payload contains no useful payload data. |
| Sync Frame Indicator | Indicates whether the frame is a synchronization frame. |
| Startup Frame Indicator | Indicates that this is a clock startup frame. |
| Frame ID | Defines the slot in which the frame is transmitted. |
| Payload Length | The number of payload bytes divided by 2. |
| Header CRC | Cyclic redundancy check of data from the sync frame indicator to the payload length. |
| Cycle Count | The value of the cycle counter. |
| Header CRC Pass | Result of Header cyclic redundancy check test. |
| Frame CRC A Pass | FlexRay frame CRC calculation algorithm - channel A. |
| Frame CRC B Pass | FlexRay frame CRC calculation algorithm - channel B. |

9.6.9 I²C protocol

I²C (Inter-Integrated Circuit) was developed by Philips in the 1980s as a low-cost way to connect peripheral chips in consumer electronics products, and is widely adopted in embedded systems. Many devices, including temperature sensors, control panels, microcontrollers, and single board prototyping systems incorporate multiple I²C interfaces for easy chip-to-chip and inter-module communications. It uses two signals: clock (SCL) and data (SDA).

| | |
|----------------------|---|
| Address Bits | Address size. |
| Address | Shown for address packets. |
| Address + R/W | Address including R/W flag. |
| R/W | Polarity of the read/write flag. |
| Address ACK | Address acknowledgment. |
| Data | The data content of the packet (hexadecimal). |
| Data ACK | Data acknowledgment. |

9.6.10 I²S protocol

I²S (Inter-IC Sound) is a serial protocol used in digital audio devices for communications between circuits such as CD transports and audio DACs. The standard was originally developed by Philips Semiconductors. It uses three signals: clock (SCK), word select (WS) and data (SD).

| | |
|----------------|--|
| Channel | 0 = channel 1 (left). 1 = channel 2 (right). |
| Data | Raw data. |
| Value | Data decoded as a two's complement value. |

9.6.11 LIN protocol

LIN (Local Interconnect Network) is a serial protocol used in automotive electronics to allow microcontrollers to communicate with low-speed peripherals. The standard was defined by the LIN Consortium, a group of five vehicle manufacturers. It uses a single wire pair with a master-slave topology.

| | |
|-----------------------------|--|
| Sync Byte | A fixed value (0x55) used for automatic baud rate detection. |
| Protected Identifier | Message address plus two parity bits. |
| Frame Identifier | Message address. |
| Parity OK | Result of Protected Identifier parity check. |
| Data | The data content of the packet. |
| Checksum | A cyclic redundancy check of the data. |
| Classic Checksum | Validates the data bytes. |
| Enhanced Checksum | Validates the data bytes and identifier. |

9.6.12 PS/2 protocol

PS/2 (Personal System/2) is the serial protocol used with PS/2 DIN connectors, designed to connect keyboards and mice to a PC. The system was developed by IBM in the late 1980s.

| | |
|------------------------|--|
| Acknowledge Bit | Device pulls 0 to acknowledge host. |
| Data | The data content of the packet. |
| Parity | Error-checking bit. Parity is odd. |
| Parity Valid | Result of parity check. |
| Data Direction | <i>Device to Host or Host to Device.</i> |

9.6.13 SENT Fast protocol

SENT (Single Edge Nibble Transmission) is a one way, point-to-point protocol for transmission of signal values from sensors to a controller. It is used mainly in automotive applications and enables reliable transmission of high-resolution data with lower cost than standards such as CAN, and higher bit rate than LIN.

This decoder extracts only the fast channel data. Use the SENT Slow decoder to extract the other channel.

For more information on the SENT Fast protocol, visit the [PicoScope A to Z](#).

| | |
|----------------------------------|---|
| Ch1 | Channel 1 data. |
| Ch2 | Channel 2 data. |
| Calibration Pulse | Duration of pulse, equal to 56 clock ticks. |
| Clock Tick | Clock period, calculated as 1/56 of calibration pulse. |
| Status | Status and communication bits. |
| Data | Variable number of nibbles (4-bit values) depending on the application. |
| Checksum | Error-checking value. |
| Reserved 1 | Reserved for specific application. |
| Reserved 2 | Reserved for specific application. |
| Serial Data | Serial data message bits. |
| Message Start | Message start = 1. Otherwise = 0. |
| Legacy Checksum Pass | Result of CRC test to February 2008 standard. |
| Recommended Checksum Pass | Result of CRC test to January 2010 standard. |

9.6.14 SENT Slow protocol

SENT (Single Edge Nibble Transmission) is a one way, point-to-point protocol for transmission of signal values from sensors to a controller. It is used mainly in automotive applications and enables reliable transmission of high-resolution data with lower cost than standards such as CAN, and higher bit rate than LIN.

This decoder extracts only the slow channel data. Use the SENT Fast decoder to extract the other channel.

For more information on the SENT Fast protocol, visit the [PicoScope A to Z](#).

| | |
|--------------------------------|--|
| 8Bit Message ID | Definition of the message ID. |
| Enhanced Serial Message | Definition of the data value. |
| CRC | The cyclic redundancy check value within the packet. |
| Configuration bit | Packet is 12-bit data and 8-bit message ID, or 16-bit data and 4-bit message ID. |
| Message ID | Message value. |
| Data | The data content of the packet. |
| Data Value | Integer value of the data field. |
| CRC Pass | Serial data message bits computed CRC matches the packet CRC. |

9.6.15 SPI protocol

SPI (Serial Peripheral Interface) bus was originally developed by Motorola for use with their microcontrollers. Due to the simplicity of the bus, other manufacturers adopted it and it has become widely available in components used in embedded system designs. It is commonly used for chip-to-chip communications between a CPU and a peripheral device such as keyboard, display, ADCs and DACs, real-time clocks, EEPROM, SD and other memory devices.

SPI is a bidirectional protocol. To decode both directions (SDI and SDO) set up two PicoScope SPI decoders, one for each direction.

For more information on the SPI protocol, visit the [PicoScope A to Z](#).

| | |
|-------------|---|
| Data | The data content of the packet (hexadecimal). |
|-------------|---|

9.6.16 UART protocol

A UART (Universal Asynchronous Receiver/Transmitter) is the communication interface found in the serial or COM ports once commonly found on computers. UARTs were developed in the 1960s for connecting modems to terminals. Each channel of data is transmitted on a single wire pair using an electrical standard such as RS-232, RS-422, or RS-485.

A typical full-duplex UART link consists of a Tx (transmit) and an Rx (receive) channel. To decode both channels, set up two PicoScope UART decoders, one for each channel.

| | |
|---------------------|---|
| Data | The data content of the packet. |
| Parity | Error-checking bit (if parity enabled). |
| Parity Check | Result of parity check (if parity enabled). |

9.6.17 USB protocol

USB (Universal Serial Bus) is widely adopted in today's personal computers and tablets for communication and power to printers, scanners, keyboard, mice, memory devices and many other peripherals. It is also used for chip-to-chip and module-to-module communication in many other types of electronic devices and embedded systems. The USB specification has evolved several times since its launch in 1996.

There are four types of USB packet: *Handshake*, *Data*, *Token*, and *Special*. *Handshake* is used to initialize communication between two end points. *Data* is used to send actual information. *Token* and *Special* (a specialized version of *Token*) are used to keep both devices informed concurrently. The data fields shown will vary depending on the USB packet type.

For more information on the USB protocol, visit picotech.com.

| | |
|----------------------|--|
| Data | The data content of the packet. |
| CRC5 | Cyclic Redundancy Check, 5 bits. |
| CRC16 | Cyclic Redundancy Check, 16 bits. |
| Pid | Packet Identifier. Determines the packet type and structure. |
| Address | For Token-specific USB packets. |
| Endpoint | For Token-specific USB packets. |
| Error Message | Set if PicoScope detects an error. |

9.7 Device feature table

Some PicoScope 6 features require specialized hardware and are therefore not available on all devices. Feature availability is indicated in the table below. (Detailed specifications for each feature may vary). Please see the relevant device data sheet for more details.

| Series / Model | 50 | AT | AW | AX | BL | DC | DI | EX | FC | FR | LP | RA | RU | SG | SM | ST | SW |
|--|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| USB DrDAQ | | | ✓ | | | | | | | | | | | ✓ | ✓ | | |
| PicoLog 1000 | | | | | | | | | | | | | | | ✓ | | |
| PicoScope 2104-2105 | | | | | | | | | | | | | | | ✓ | | |
| PicoScope 2202 | | | | | | | | | | | | | | | ✓ | | |
| PicoScope 2203 | | | ✓ | | | | | | | | | | | ✓ | ✓ | | ✓ |
| PicoScope 2204-2205 | | ✓ | ✓ | | | | | | | | | | | ✓ | ✓ | | ✓ |
| PicoScope 2206-2208 | | ✓ | ✓ | | | ✓ | | ✓ | | | | ✓ | | ✓ | ✓ | ✓ | ✓ |
| PicoScope 2204A | | ✓ | ✓ | | | | | | | | | | | ✓ | ✓ | | ✓ |
| PicoScope 2205A | | ✓ | ✓ | | | | | | | | | | | ✓ | ✓ | | ✓ |
| PicoScope 2206A | | ✓ | ✓ | | | ✓ | | | | | | ✓ | | ✓ | ✓ | ✓ | ✓ |
| PicoScope 2207A | | ✓ | ✓ | | | ✓ | | | | | | ✓ | | ✓ | ✓ | ✓ | ✓ |
| PicoScope 2208A | | ✓ | ✓ | | | ✓ | | | | | | ✓ | | ✓ | ✓ | ✓ | ✓ |
| PicoScope 2205 MSO | | ✓ | ✓ | | | | ✓ | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 2205A MSO | | ✓ | ✓ | | | ✓ | ✓ | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 2000B | | ✓ | ✓ | | | ✓ | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 2000B MSO | | ✓ | ✓ | | | ✓ | ✓ | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 2405A | | ✓ | ✓ | | | ✓ | | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 3204 | | | | | | | | ✓ | | | | | | ✓ | | | |
| PicoScope 3205-3206 | | | | | | | | ✓ | | | ✓ | | | ✓ | | | ✓ |
| PicoScope 3223/3423 | | ✓ | | | | | | | | | | | | | ✓ | | |
| PicoScope 3224/3424 | | ✓ | | | | | | | | | | | | | ✓ | | |
| PicoScope 3425 | | ✓ | | | | | | | | | | | | | ✓ | | |
| PicoScope 3200A | | ✓ | | | | ✓ | | ✓ | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 3400A | | ✓ | | | ✓ | ✓ | | ✓ | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 3200B | | ✓ | ✓ | | | ✓ | | ✓ | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 3400B | | ✓ | ✓ | | ✓ | ✓ | | ✓ | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 3000D | | ✓ | ✓ | | ✓ | ✓ | | ✓ | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 3000 MSO | | ✓ | ✓ | | | ✓ | ✓ | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 3000D MSO | | ✓ | ✓ | | ✓ | ✓ | ✓ | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 4223-4224 | | ✓ | | | | | | | ✓ | | ✓ | ✓ | ✓ | | ✓ | | |
| PicoScope 4423-4424 | | ✓ | | | | | | | ✓ | | ✓ | ✓ | ✓ | | ✓ | | |
| PicoScope 4225/4425 | | ✓ | | | ✓ | | | | ✓ | | ✓ | ✓ | ✓ | | ✓ | | |
| PicoScope 4226-4227 | | ✓ | ✓ | | | | | ✓ | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 4262 | | ✓ | ✓ | | ✓ | | | ✓ | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 4824 | | ✓ | ✓ | | | ✓ | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 5203-5204 | | ✓ | ✓ | ✓ | | | | ✓ | | | ✓ | ✓ | | ✓ | ✓ | ✓ | ✓ |
| PicoScope 5000A | | ✓ | | | ✓ | ✓ | | ✓ | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 5000B | | ✓ | ✓ | | ✓ | ✓ | | ✓ | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 6000 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 6000A/C | ✓ | ✓ | | ✓ | ✓ | ✓ | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 6000B/D | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| PicoScope 6407 | ✓ | ✓ | ✓ | ✓ | | | | | | | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

50 [50 ohm inputs](#)
 AT [Advanced triggers](#)
 AW [Arbitrary waveform generator](#)
 AX [Auxiliary input/output](#)
 BL [Switchable bandwidth limiter](#)
 DC [DC offset adjustment](#)
 DI [Digital inputs](#)

FR [Flexible resolution](#)
 LP [Lowpass filtering](#)
 RA [Rapid triggering](#)
 RU [Runt pulse trigger](#)
 SG [Signal generator](#)
 SM [Streaming mode](#)
 ST [Signal generator triggers](#)

EX [External trigger input](#)
FC [Frequency counter](#)

SW [Signal generator sweep modes](#)

9.8 Command-line syntax

PicoScope can be run from the Windows command line, allowing you to run tasks manually or under the control of a batch file or other program.

To display the GUI

```
PicoScope <filename>
```

<filename> Specifies a single .psdata or .pssettings file.

Example: `PicoScope C:\Temp\source.psdata`

To display help

```
PicoScope /?
```

Shows help on all command-line options.

To convert a psdata file

```
PicoScope /C,/c
```

Converts a psdata file from one format to another. Cannot be used with `/p[rint]`.

Syntax:

```
PicoScope /c[onvert] <names> [/d <names>] /f <format> [/q]
[/b [<n>[:<m>]] | [all]] [/v <viewportname>]
```

<names> Specifies a list of one or more directories or psdata files. Wildcards may be used to specify multiple files. If a directory is specified, all psdata files within that directory will be specified. This is a mandatory argument.

/d <names> Destination. Default is input filename with new extension.

/f <format> Destination format: csv, txt, png, bmp, gif, agif [animated GIF], psdata, pssettings, mat [MATLAB]. This is a mandatory argument.

/q Quiet mode. Do not ask before overwriting files. The default is to prompt.

/b [<n>[:<m>]]|all Waveform number n, waveform range n to m or all waveforms. The default is the current waveform.

/v <viewportname> View to convert. Default is current view.

Example:

```
PicoScope /c C:\Temp\source.psdata /f png /b 5:9 /v Scope2
```

To print a view

```
PicoScope /P,/p
```

Prints a view in the psdata file. Cannot be used with `/c[onvert]`.

Syntax:

```
PicoScope /p[rint] <names> [/b [<n>[:<m>]] | all] [/v <viewportname>]
```

`<names>` Specifies a list of one or more directories or .psdata files. Wildcards may be used to specify multiple files. If a directory is specified, all .psdata files within that directory will be specified. This is a mandatory argument.

`/b [<n>[:<m>]]|all` Waveform number `n`, waveform range `n` to `m` or all buffers. Default is current waveform.

`/v <viewportname>` View to convert. Default is current view.

Example:

```
PicoScope /p C:\Temp\source.psdata /b 5:9 /v Scope2
```

To import notes

```
PicoScope /N,/n
```

Copies text from a specified file to the [Notes area](#).

Syntax:

```
PicoScope /n[otes] <notes filename> <filename>
```

`<notes filename>` Specifies a single text file.

`<filename>` Specifies a single psdata or pssettings file.

Example:

```
PicoScope /n C:\Temp\source.txt C:\Temp\source.psdata
```

To run an automation command

```
PicoScope /A,/a
```

Run automation command or macro on an existing instance of PicoScope 6.

Syntax:

```
PicoScope /a[utomation] <command> | <macro>
```

`<command>` Automation command

`<macro>` Path to .psmacro file containing a [macro](#)

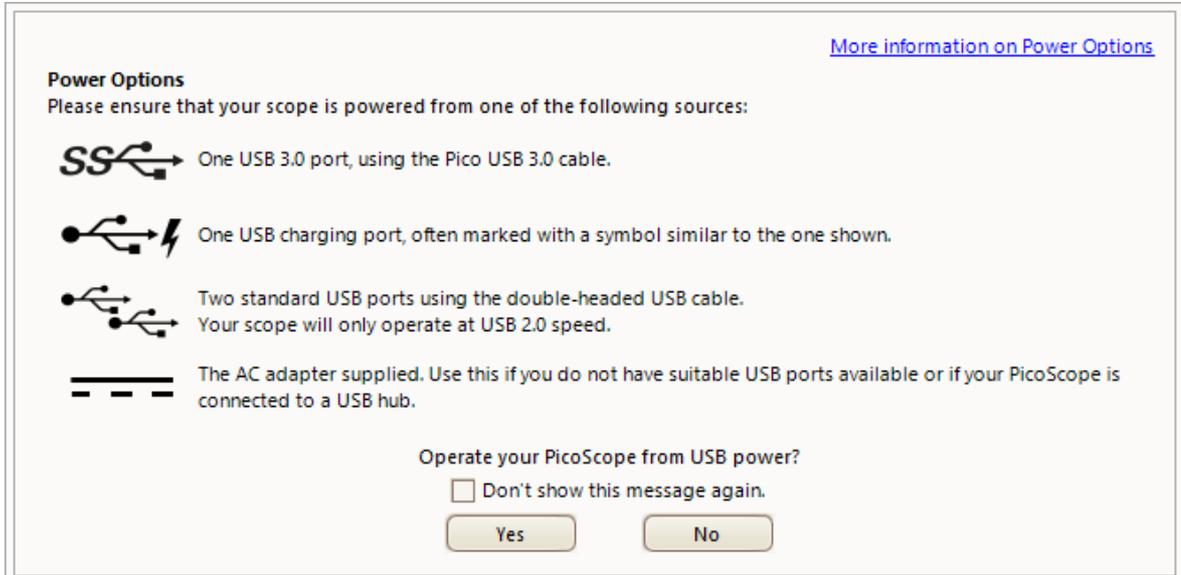
Examples:

```
PicoScope /a Run.Pressed=True  
PicoScope /a MyMacro.psmacro  
PicoScope /a ? (prints a list of automation commands to the console)  
PicoScope /a Measurements? (prints information on the Measurements command)
```

An instance of PicoScope must be running before any automation command is issued.

9.9 Flexible power

The flexible power system for PicoScope devices provides a choice of power sources. In most cases a straightforward USB connection will be enough to power the scope. If PicoScope needs you to switch to a different powering method, it will display a dialog similar to this:

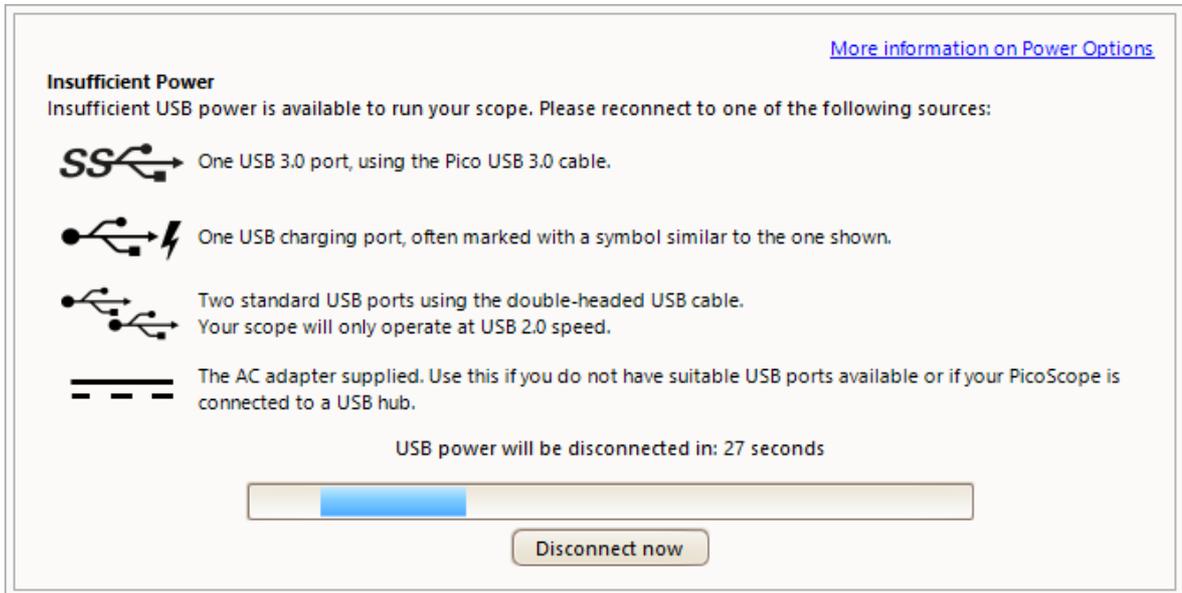


The power icons explained

-  **USB 3.0 port.** If this symbol appears, you may power the scope from any USB 3.0 port using the USB 3.0 cable provided with your scope.
-  **USB charging port.** If this symbol appears, you may power the scope from any USB port capable of supplying 1200 mA of current.
-  **Dual USB port power.** If this symbol appears, you may power the scope from any two powered USB ports using the dual-headed USB 2.0 cable supplied with your scope.
-  **AC adapter.** Use the adapter supplied and plug it into the **DC IN** socket on your scope. If you power the scope in this way, the **Power Options** dialog will automatically close and the scope will start working immediately.

Insufficient USB power

If insufficient USB power is available, PicoScope displays this dialog:



Connect one of the power sources listed and try again. If you plug in the AC adapter, the dialog will automatically close and you can continue working.

9.10 Glossary

AC coupling. In this mode, the scope device rejects very low signal frequencies below about 1 hertz. This allows you to use the full resolution of the scope to measure a.c. signals accurately, ignoring any DC offset. You cannot measure the signal level with respect to ground in this mode.

AWG. An arbitrary waveform generator (AWG) is a circuit that can generate a waveform of almost any shape. It is programmed with a data file, supplied by the user, which defines the output voltage at a number of equally spaced points in time. The circuit uses this data to reconstruct the waveform with a specified amplitude and frequency.

Axis. A line marked with measurements. PicoScope shows one vertical axis for each channel that is enabled in a view, giving measurements in volts or other units. Each view also has a single horizontal axis, which is marked in units of time for a scope view, or units of frequency for a spectrum view.

Channel. A scope device has one or more channels, each of which can sample one signal. High-speed scope devices typically have one BNC connector per channel.

CSV. Comma-separated values. A text file containing tabulated data, with columns separated by commas and rows by line-breaks. CSV format is used for importing and exporting PicoScope [arbitrary waveform files](#). You can also export PicoScope waveforms in CSV format. CSV files can be imported into spreadsheets and other programs.

DC coupling. In this mode, the scope device measures the signal level relative to signal ground. This shows both DC and AC components.

Dead time. The time between the end of one capture and the start of the next. To obtain the minimum possible dead time, use **Rapid** trigger mode.

Demo mode. If PicoScope is started when no scope device is plugged in, it allows you to select a "demo device", a virtual scope unit that you can use to test the software. The program is then in **demo** (short for *demonstration*) mode. This mode provides a simulated, configurable signal source for each input channel of the demo device.

ETS. Equivalent Time Sampling. A method of increasing the effective sampling rate of the scope. In a scope view, the program captures several cycles of a repetitive signal, then combines the results to produce a single waveform with higher time-resolution than a single capture. For accurate results, the signal must be perfectly repetitive and the trigger must be stable.

Floating inputs. A feature of the PicoScope 4225 and 4425 oscilloscopes. These are inputs that do not share a common measurement ground. The channels are separated by high impedances and their measurement grounds can be connected to any voltage within the specification range. It is important however that every input in use has both a signal and a ground connection.

Graticule. The horizontal and vertical dashed lines in every view. These help you estimate the amplitude and time or frequency of features on the waveform.

Grid. The arrangement of viewports. The number of grid rows and the number of grid columns can each be either 1, 2, 3 or 4.

IEPE. Integrated circuit piezoelectric. A type of sensor, usually built to detect acceleration, vibration or sound, and with a built-in amplifier. IEPE sensors can only be used with special PicoScope oscilloscopes that have IEPE-compatible inputs.

In focus. PicoScope can display several views, but only one view is in focus at any time. When you click a toolbar button, it will usually affect only the view that is in focus. To bring a view into focus, click on it.

MSO. Mixed-signal oscilloscope. An instrument that captures and displays analog and digital signals on the same timebase.

PC Data Logger. A measuring instrument consisting of a hardware interface and the PicoLog software running on a PC. You can also use the device with the PicoScope software to create a multi-channel voltage input oscilloscope.

PC Oscilloscope. A measuring instrument consisting of a [scope device](#) and the PicoScope software running on a PC. A PC Oscilloscope has the same functions as a traditional bench-top oscilloscope but is more flexible and cost-effective. You can improve its performance by upgrading the PC using standard parts from any computer shop, or by purchasing a new scope device, and you can upgrade the software by downloading an update from Pico Technology.

Probe. An accessory that attaches to your oscilloscope and picks up a signal to be measured. Probes are available to pick up any form of signal, but they always deliver a voltage signal to the oscilloscope. PicoScope has built-in definitions of standard probes, but also allows you to define custom probes.

Progressive mode. Normally, PicoScope redraws the waveform in a scope view many times every second. At timebases slower than 200 ms/div, however, it switches to progressive mode. In this mode, PicoScope updates the scope view continuously as each capture progresses, rather than waiting for a complete capture before updating the view.

Resolution enhancement. Collecting samples at a faster rate than requested, then combining the excess samples by averaging. This technique can increase the effective resolution of a scope device when there is a small amount of noise on the signal. ([More details](#)).

Ruler. A vertical or horizontal dashed line that can be dragged into place on a waveform in a view. PicoScope displays the signal level, time value or frequency value of all rulers in the Ruler Legend box.

Scope device. The box from Pico Technology that you plug into the USB or parallel port of your computer. With the help of the PicoScope software, the scope device turns your computer into a PC Oscilloscope.

Standard deviation. A statistical measure of the spread of a set of samples. The standard deviation of the set $y_0 \dots y_{n-1}$ is defined as:

$$SD = \sqrt{\frac{1}{(n-1)} \sum_{i=0}^{n-1} (y_i - \bar{y})^2}$$

where \bar{y} is the arithmetic mean of all the samples. The units of the standard deviation value are the same as those of the original samples.

Tool tip. A label that appears when you move the mouse pointer over some parts of the PicoScope screen, such as buttons, controls and rulers.

Trigger. The part of an oscilloscope that monitors an incoming signal and decides when to begin a capture. Depending on the trigger condition that you set, the scope may trigger when the signal crosses a threshold, or may wait until a more complex condition is satisfied.

Vertical resolution. The number of bits that the scope device uses to represent the signal level. This number depends on the design of the device, but can be boosted in some cases by using [resolution enhancement](#).

View. A presentation of data from a scope device. A view may be a [scope view](#), an [XY view](#) or a [spectrum view](#).

Viewport. The views in the [PicoScope window](#) are arranged in a [grid](#), and each rectangular area in the grid is called a viewport.

Index

%

- %buffer% variable 90
- %file% variable 90
- %time% variable 90

5

- 50 Ω DC inputs 117, 202

A

- AC adapter 207
- AC coupling 208
- AC power 98
- Accelerometer inputs 117
- Access 3
- Add Measurement dialog 53
- Adding a measurement 53
- Advance function 77
- Advanced measurement settings 54
- Advanced Options toolbar
 - Channel Labels button 116
 - Notes button 116
 - Rulers button 116
- Advanced triggering 153, 155, 202
 - types 156
- Alarms 34, 89
 - save on trigger 178
- Analog intensity 137
- Analog offset 119
- AND logical operator 161
- Animated GIF 39
- Arbitrary waveform generator 141, 202, 208
 - editing window 146
 - files 145
 - importing from channel 149
- Automotive menu 111
- Auxiliary (AUX) I/O 156, 202
- Average (statistics) 20
- AWG 208
- Axis 11, 16, 18, 208
 - auto-arrange 51
 - horizontal 11, 16, 18
 - offset 169
 - scaling 119, 121, 169
 - vertical 11, 16, 18

B

- Bandwidth limiter 119, 202
- Battery power 98
- Beep 89
- Binary files, exporting 42
- Bit stream 147
- bmp files 39
- Bring channel to front 121
- Buffer Navigation toolbar 139
- Buffer Overview 35

C

- Capture count 20
- Capture modes 8, 9
- Capture rate preference 98
- Capture Setup toolbar 133
- Capture size preference 97
- Changing device 164
- Channel 208
 - selecting in a view 50
- Channel Labels 48, 116
- Channel Options
 - button 117
 - menu 118
- Channel ordering 121
- Channel settings
 - in Properties sheet 28
- Channels toolbar
 - PicoLog 1000 Series 128
 - standard 117
 - USB DrDAQ 130
- Clipboard 47
- Collection time units 97
- Color preferences 105
- Column auto-width 52
- Command-line syntax 204
- Connect Device dialog 37, 112
- ConnectDetect 124
- Converting data files 114, 204
- Copy
 - as image 47
 - as text 47
- Copyright 3
- Coupling control 117
- Create New Custom Probe dialog 57
- CSV files 39, 208
 - exporting 41
- Cursors (see Rulers) 22, 23, 27
- Custom grid layout dialog 51

- Custom Probe Manager 55
- Custom Probe wizard 57
 - Create New Custom Probe dialog 57
 - Edit Existing Custom Probe dialog 58
 - Edit Range dialog 64
 - Edit Range dialog (Advanced tab) 66
 - Finished dialog 69
 - Lookup-table Scaling dialog 61
 - Manual Ranges Setup dialog 63
 - Probe Identification dialog 68
 - Probe Output Units dialog 59
 - Range Management dialog 62
 - Scaling Method dialog 60
- Custom probes 29
 - dialog 56
 - saving 37

D

- Data files
 - converting 114
- DC coupling 208
- DC IN power 207
- DC offset 119, 202
- Dead time 208
- Decoding tab 32
- Default Print Settings 104
- Delay function 77
- Demo device 113, 150
- Demo mode 113, 150, 151, 209
- Demo Signals
 - dialog 151
 - menu 150
- Details dialog
 - channel labels 49
 - customer details 49
 - vehicle details 49
- Device feature table 202
- Device, how to change 164
- Digital color 137
- Digital inputs 126, 202
- Digital Inputs button 117, 125
- Digital outputs 129
 - USB DrDAQ 132
- Digital Setup dialog 126
- Digital trigger dialog 159
- Digital view 14
 - context menu 15
- DrDAQ 130
- Dropout trigger 157

E

- Edge trigger 156
- Edit Existing Custom Probe dialog 58
- Edit menu 47
- Edit Range dialog 64
 - Advanced tab 66
- Edit Reference Waveform dialog 82
- Effective resolution 119
- Equation dialog 74
- Equivalent time sampling 153
- ETS 153, 209
 - and Advanced Triggering 155
- Exit 38
- Exporting data 41
 - binary format 42
 - text format 41
- External (EXT) trigger 156, 202

F

- Fall time
 - threshold 54
- Falling edge button 154
- Feedback 110
- File conversion 114
- File menu 37
- Filter Method dialog 67
- Filtering 118
 - channels 122
 - measurements 20
 - statistics 54
- Fitness for purpose 3
- Flexible power 207
- Flexible Resolution 202
- Focus 209
- Frequency counter 117, 202
- Frequency difference, measuring 166
- Frequency legend 23, 27
- Frequency rulers 18, 23
- Functions, mathematical 30, 74

G

- General preferences 96
- Generate Mask
 - dialog 93
- Generate mask button 91
- gif files 39
- Glitches, finding 156
- Glossary 208
- Graticule 11, 16, 18, 209

Grid 209
 layout 51
 Groups, digital input 127

H

Hand tool 162
 Harmonic control for measurements 54
 Help menu 110
 Horizontal axis 11, 16, 18
 House current 98
 Hysteresis 158

I

IEPE 209
 inputs 117
 Image, saving as 38
 Import from a Channel dialog 149
 Input impedance 117
 Interpolation
 linear 100
 sin(x)/x 100
 Interval trigger 157
 Invert button 30

K

Key features 2
 Keyboard maps 102
 Keyboard shortcuts 101, 162

L

Language preferences 103
 LED
 on USB DrDAQ 131
 Legal statement 3
 Liability 3
 Line thickness 106
 Link file 87
 creating 182
 Lissajous figures 16
 Logic trigger 157
 dialog 160
 Lookup-table Scaling dialog 61
 Lowpass filtering 67, 118, 122, 202

M

Macro
 Recorder 94
 running from command line 205

Mains power 98
 Manual Ranges Setup dialog 63
 Marquee Zoom tool 162
 Mask Limit Testing 33, 55
 How to set up 175

Masks

colors 33, 105
 displaying 51
 editing 92
 exporting 91
 generating 91
 importing 91
 in Buffer Overview 35
 mask files 91
 Mask Library dialog 91
 menu 91
 polygons 92

Math Channel Wizard

Equation dialog 74
 Finished dialog 80
 Introduction dialog 73
 Name and Color dialog 78
 overview 72
 Units and Range dialog 79

Math channels 55, 70

Built-in 70
 dialog 70
 Library 70
 Loaded 70
 overview 30
 saving 37
 warning symbol 30

MATLAB files

exporting 42, 82
 saving 39

Max (statistics) 20

Maximum Waveforms preference 96

Measurement system

selecting 103

Measurements

adding 20, 52, 53
 advanced settings 54
 capture size 97
 deleting 20, 52
 editing 20, 52
 font size 52
 list of types 184
 menu 52
 scope 184
 spectrum 185
 statistics 20
 table 20

- Measurements
 - toolbar 140
- Menus 36
- Metric measurements 103
- Min (statistics) 20
- Missing events, finding 157
- Mission-critical applications 3
- MSO 209
 - setup 126
 - view 13

N

- NAND logical operator 161
- NOR logical operator 161
- Normal Selection tool 162
- Notes 48, 116
 - importing from command line 205
 - notes area 47

O

- Offset 168
 - analog 119
- Open file 37
- OR logical operator 161
- Oscilloscope 6
- Overrange
 - BNC shell to chassis 12
 - indicator 11, 117
 - normal measurement range 12

P

- Page Down key 37
- Page Up key 37
- Panning 163
- Pattern trigger 159
- PC Data Logger 209
- PC Oscilloscope 7, 209
- Persistence mode 19
 - button 133
 - enabling and disabling 107
 - options 137
- Phase rulers 24
 - partition 26
 - units 26
 - wrap 26
- PicoLog 1000 Series 128, 129
- PicoScope 6 1, 2, 7
 - how to use 3, 5, 6
 - main window 10
- png files 39

- Pointer tool tip 21
- Polygon 92
- Post-trigger delay 189
 - arrow 17
 - control 154, 189
- Power Management preferences 98
- Power supply 207
- Preferences 55
 - capture rate 98
 - colors 105
 - Default Print Settings 104
 - device selection 107
 - dialog 95
 - general 96
 - keyboard 101
 - language 103
 - persistence modes 107
 - power management 98
 - sampling 99
 - spectrum modes 107
 - updates 109
- Pre-trigger delay 189
 - control 154, 189
- Printing 37
 - from command line 205
 - from menu 37
 - preferences 104
 - preview 37
- Probe 209
 - custom 29
 - Identification dialog 68
 - Output Units dialog 59
- Progress bar 136
- Progressive mode 209
- Properties sheet 28
 - displaying 51
- psdata files
 - converting 114, 204
 - saving 38
- pskeys files 102
- psmaths files 71
- psreference files 82
- pssettings files 39
- Pulse width trigger 156
- PWM output
 - PicoLog 1000 Series 129
 - USB DrDAQ 132

R

- Range control 117
- Range Management dialog 62

- Rapid triggering 153, 202
 - Reference waveforms 55
 - adding 51
 - delay 121
 - dialog 81
 - Library 81
 - Loaded 81
 - overview 31
 - using in equations 75
 - Reset 'Don't show this again' dialogs 96
 - Resolution control 134
 - Resolution enhancement 118, 119, 210
 - RGB LED on USB DrDAQ 131
 - Rise time
 - threshold 54
 - Rising edge button 153
 - Rotation rulers 24
 - partition 26
 - units 26
 - wrap 26
 - RPM 27, 108
 - Rulers 11, 18, 116
 - definition 210
 - deleting 22, 23
 - frequency 18
 - handles 11, 22, 23, 24
 - legend 27
 - lock button 27
 - phase 24
 - rotation 24
 - settings 26
 - signal 11, 18
 - time 11, 23
 - Runt pulse trigger 157, 202
- ## S
- Sampling preferences 99
 - Sampling rate 134
 - Save As 37
 - dialog 38
 - Save file 37
 - Save on trigger 90, 178
 - Scaling 121, 168
 - button 7
 - Scaling Method dialog 60
 - Scope device 210
 - Scope measurements
 - AC volts 184
 - Cycle time 184
 - DC volts 184
 - Duty cycle 184
 - Fall time 184
 - Falling rate 184
 - Frequency 184
 - High pulse width 184
 - Low pulse width 184
 - Maximum 184
 - Minimum 184
 - Peak to peak 184
 - Rise time 184
 - Rising rate 184
 - Scope mode 8
 - button 133
 - Scope view 9, 11
 - Scrolling 168
 - Selection tool, normal 162
 - Send channel to back 121
 - Serial decoding 32, 55
 - data window 87
 - dialog 83
 - link file 87
 - protocols 191
 - Serial number
 - of oscilloscope 110
 - Settings
 - saving 37
 - Signal difference, how to measure 165
 - Signal generator
 - button 141
 - dialog 141
 - sweep mode 142, 202
 - triggering 202
 - USB DrDAQ 144
 - Waveform types 187
 - Signal rulers 11, 18, 22
 - Sin(x)/x preferences 100
 - Slow sampling transition 99
 - Smoothing 100
 - Software version 1
 - Sound file 90
 - Space bar 152
 - Spectrum measurements
 - Amplitude at peak 185
 - Frequency at peak 185
 - Intermodulation Distortion (IMD) 186
 - Signal to Noise Ratio (SNR) 185
 - Spurious-free Dynamic Range (SFDR) 185
 - Total Harmonic Distortion (THD) 185
 - Total Harmonic Distortion plus Noise (THD+N) 185
 - Total power 185
 - Spectrum mode 8
 - button 133

- Spectrum mode 8
 - enabling and disabling 107
- Spectrum options
 - bins 135
 - dialog 135
 - display mode 136
 - scale 136
- Spectrum view 9, 18
 - how to set up 170
- Spreadsheet, exporting to 39
- Spurious edges, finding 157
- Standard deviation 20, 210
- Start / Stop toolbar 152
- Startup Settings menu 45
- Statistics 20
 - filtering 54
- Support 3
- Sweep mode 142, 202
- Symbols
 - red warning 11, 12, 117
 - yellow warning 12
- System requirements 4

T

- Text files, exporting 39, 41
- Threshold for measurements 54
- Thresholds, digital input 126
- Time difference, how to measure 166
- Time gate 28
- Time rulers 11, 23
- Timebase controls 133
- Tool tip 210
- Toolbars 116
 - move bottom toolbar 108
- Tools menu 55
- Trace 6
- Tracking rulers 27
- Trade marks 4
- Trigger 153, 189, 210
 - advanced 153, 155
 - digital 159
 - dropout 157
 - dual-edge 156
 - edge 156
 - glitches 156
 - interval 157
 - logic 157
 - marker 17
 - missing events 157
 - mode control 153
 - pulse width 156

- reference point 189
- runt pulse 157
- timing 189
- toolbar 153
- window 156
- txt files 39

U

- U.S. measurements 103
- Undo zoom 163
- Updates 110
- Upgrades 3
- Usage 3
- Usage statistics 109
- USB DrDAQ 130
- USB power 207
- 'Used' check box 160
- User's Guide 110

V

- Valid edges, finding 157
- Version number
 - hardware 110
 - software 1, 110
- Vertical axis 11, 16, 18
- Vertical resolution 210
- View 210
 - enabling sub views 50
 - how to move 167
 - menu 50
 - scope 11
 - selecting channels 50
 - spectrum 18
 - XY 16
- Viewport 210
- Viruses 3

W

- Warning symbol
 - red 11, 12, 117
 - yellow 12
- Waveform 6, 11
 - saving 37
- Waveform buffers
 - number of 96
- Window functions 135, 188
- Window trigger 156

X

- X axis, configuring 50
- X-Axis command 16
- XNOR logical operator 161
- XOR logical operator 161
- XY view 16

Z

- Zero offset 119
- Zooming 168
 - undo 163
 - Zoom Overview window 163
 - Zooming and Scrolling toolbar 162

United Kingdom headquarters

Pico Technology
James House
Colmworth Business Park
St. Neots
Cambridgeshire
PE19 8YP
United Kingdom

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

sales@picotech.com
support@picotech.com

www.picotech.com

United States headquarters

Pico Technology
320 N Glenwood Blvd
Tyler
Texas 75702
United States

Tel: +1 800 591 2796
Fax: +1 620 272 0981