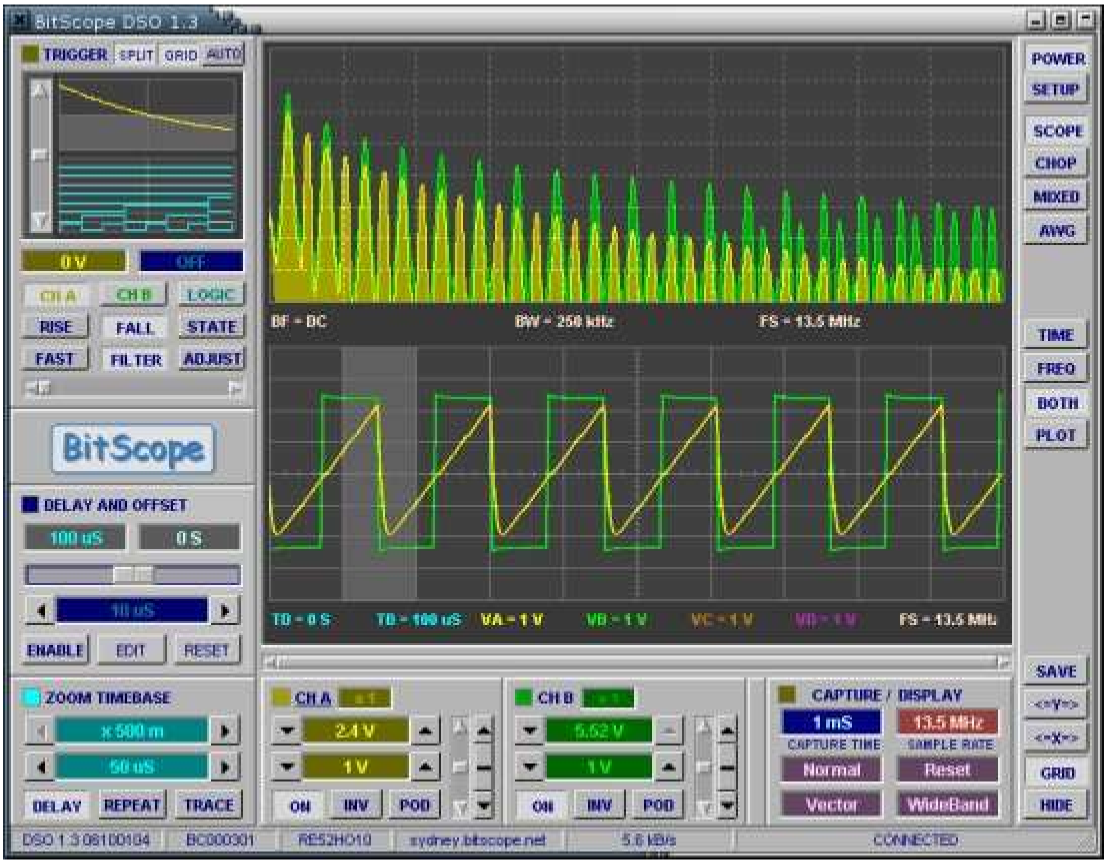


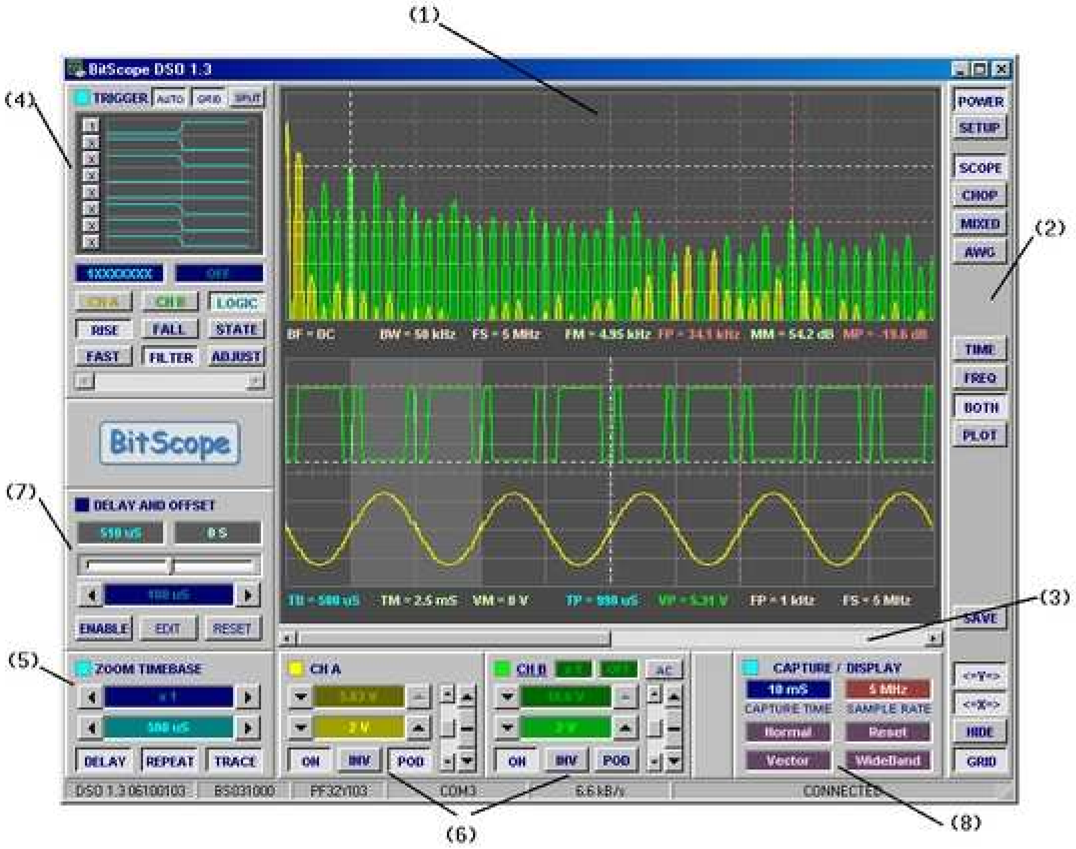
BitScope DSO 1.3 User Guide

Revision 1.00, October 16, 2006.

Suite G03 / 28 Chandos Street., St. Leonards NSW 2065 AUSTRALIA
Phone: +61 2 9436 2955 Fax: +61 2 9436 3764
Email: sales@bitscope.com Web: www.bitscope.com

DSO 1.3 ONLINE GUIDE

BitScope DSO is the premier software application for all BitScopes



BitScope DSO Application (click for a larger view)

The screenshot above shows the default layout (when run on a **BS310**)

The main features of this software are:

(1) Main Display: waveforms, logic, **spectrum**, scope **variables** and **cursors**.

(2) Instrument Menus: select DSO instruments, display modes and cursors.

(3) Waveform Offset: used to scroll the waveform and/or logic traces left and right.

(4) Trigger Control: controls trigger setup and displays trigger waveform and data.

(5) Zoom Timebase: primary timebase and timebase expansion (aka zoom) control.

(6) Channel Controls: controls input source, range, vertical position and scaling.

(7) Delay and Offset: delay timebase and post-trigger delay, reports waveform offset.

(8) Capture Control: capture sample rate, duration, frame rate and display modes.

The following pages explain in more detail the capabilities of this software.

SETUP AND STARTUP

BitScope DSO setup is simple (and usually unecessary)

When the DSO first starts up it displays a lissajou figure in the main display and two buttons:

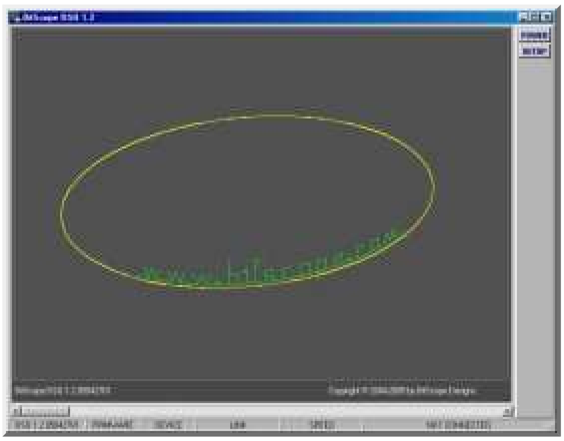
(1) POWER: click to "switch on the DSO".

(2) SETUP: click to open the setup dialog.

DSO supports USB, Ethernet and RS-232 links to BitScope devices and the setup dialog is where you tell the DSO which one to use.

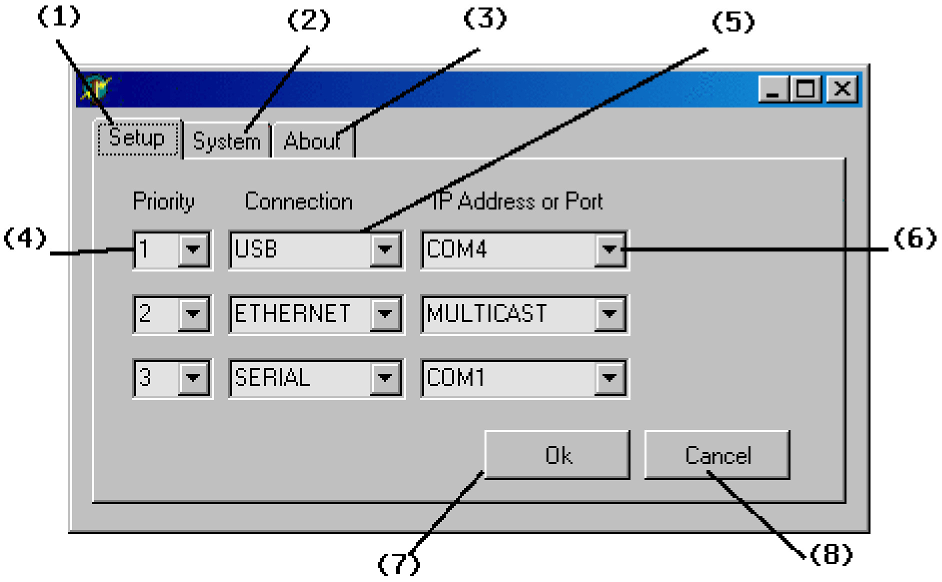
There are built-in defaults which attempt to connect on COM ports from 1 to 9 and also Ethernet (if your PC is network connected).

You only need to run setup if the DSO cannot find your BitScope.



DSO Power On Screen

DSO is a test instrument which means most settings are accessed directly on screen or via pop-ups rather than via nested menus or dialog boxes. One exception is the SETUP dialog:



Priority	Connection	IP Address or Port
1	USB	COM4
2	ETHERNET	MULTICAST
3	SERIAL	COM1

(1) Setup Page: selects the BitScope connection preferences.

(2) System Page: configures system defaults for the DSO (not active in 1.3).

(3) About Page: some information about the DSO and its authors.

(4) Connection Priority: establishes the priority of this connection.

(5) Protocol Type: selects SERIAL, USB or ETHERNET connection methods.

(6) Address or Port: selects serial or USB port or IP connection address. This pop-down menu shows a set of common default values. However if the value required is not shown you can simply type it in (eg, COM10).

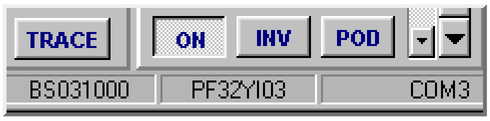
(7) Accept Change: accepts changes and saves them to the probe file.

(8) Cancel Change: close the setup dialog without making any changes.

The setup lists three configurable connection methods.

The DSO attempts connects to your BitScope using one of these methods.

To do this, it searches the list in decending priority order until it finds the connected BitScope.
Note: the number of connection specifications can in fact be more than three but to do this you must manually edit the DSO's probe file.



When the connection is made, it is indicated at the bottom of the DSO as this example shows.

Here a BS310U identified as PF32YI03 has been located at COM Port 3 (in this case, a USB COM Port).

If the UDP/IP protocol was used to connect with a Network BitScope it would report the IP address (or domain name) of the BitScope to which it is connected.

If no BitScope can be found the DSO defaults to OFFLINE REPLAY MODE which means the DSO is running but is not connected. In this case a file dialog appears asking for a **DDR file** to play.

About the Lissajou display...

The lissajou display shown when the DSO is running but not "powered on" updates at about 5 fps and provides a handy performance test of your PC. If the lissajou updates much slower than this, or if your PC becomes sluggish when it is running, your PC or graphics card may need some tweaking to work well with DSO.

For a bit of fun, try moving the slider at the bottom of the screen :-)

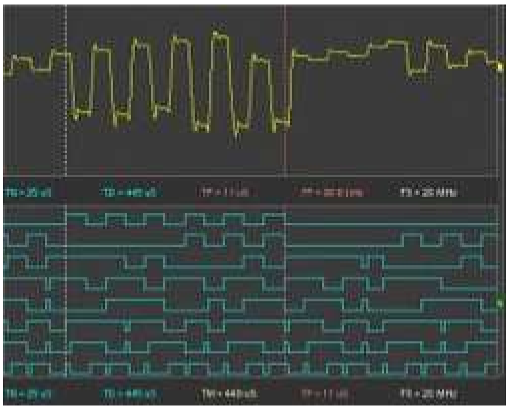
VIRTUAL INSTRUMENTS

BitScope DSO comprises an integrated set of Virtual Instruments

A unique feature of the DSO is that while its name may imply "Digital Storage Oscilloscope" it is in fact a set of integrated "virtual instruments" that can do far more than an normal oscilloscope.

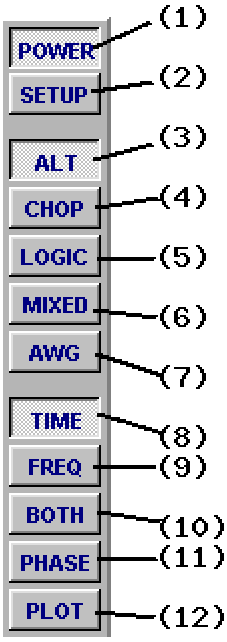
DSO Version 1.3 supports the following test, measurement and data acquisition instruments:

- **Digital Storage Oscilloscope**
- **Mixed Signal Oscilloscope**
- **Logic State Analyzer**
- **Waveform Generator**
- **Spectrum Analyzer**
- **Data Recorder**



All of these instruments share access to the BitScope to which the DSO is connected.

They are also integrated with each other to allow comparative measurements to be made between different instruments and different capture modes of each instrument.



The control bar down the right side of the DSO application window is where the instrument, display and **global controls** are located.

The buttons that appear depend on the BitScope and selected instrument.

For example, on **BS50U** the buttons appearing with ALT selected are:

(1) **POWER**: turns the DSO "on" and "off".

(2) **SETUP**: BitScope connection preferences.

(3) **ALT**: ALT capture dual channel mode.

(4) **CHOP**: CHOP capture dual channel mode.

(5) **LOGIC**: logic analyzer capture mode.

(6) **MIXED**: mixed analog/logic capture mode.

(7) **AWG**: arbitrary waveform generator mode.

(8) **TIME (waveform)**: waveform display display.

(9) **FREQ (frequency)**: magnitude spectrum display.

(10) **BOTH (wave/freq)**: waveform and spectrum (sppt) display.

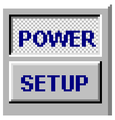
(11) **PHASE (frequency)**: magnitude and phase spectrum display.

(12) **PLOT (waveform)**: plot waveform (CHA vs CHB) display.

Power and Setup

(1) (2)

POWER and SETUP are always present.



POWER: click to "switch on the DSO".

SETUP: click to open the setup dialog.

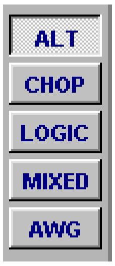
More information about setup and configuration is available **here**.

Instrument Selectors

(3)...(7)

The selected instrument configures the operation of DSO for different types of measurement. Usually you choose the most appropriate instrument as the first step when performing a new measurement.

With BS2xx and BS3xx BitScopes the choices are **ALT** (alternating capture), **CHOP** (chopped capture) and MIXED (mixed signal capture).



On BS4xx CHOP and ALT are merged into one instrument (called SCOPE) and on BS5x an additional logic only instrument (called LOGIC) is available.

BitScope models that include a waveform generator also include the AWG instrument.

ALT is usually preferred for single or dual channel high bandwidth analog capture and CHOP for lower bandwidth phase aligned dual channel analog capture

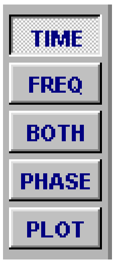
LOGIC is for for high speed logic only capture, MIXED for analog + logic capture, and AWG for waveform generation (and simultaneous waveform capture in the case of BS310).

Display Selectors

(8)...(12)

Below the instrument selectors are the display selectors which vary according to the instrument.

SCOPE, ALT, CHOP and AWG instruments have choices TIME, FREQ, BOTH, PHASE and PLOT.



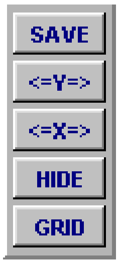
The MIXED instrument has choices MIXED, LOGIC and TIME.

The display selection is "modal". That is, if you change the selected instrument, the display (and other settings like timebase) most recently selected for that instrument will be restored.

Display selectors do not affect the way data is captured, only how it is presented on the display. They configure the X and Y axes, scaling and measurement units, graticule type, cursors and splits.

Global Controls

Below the display selectors are several global controls:



SAVE exports the current display image as a bitmap file (BMP).

The resulting graphic may be imported as an image in word documents or encoded as PNG, GIF or JPEG files for use on the web or as email attachments.

The **<=X=>** and **<=Y=>** buttons enable the display **cursors**.

HIDE flips the application between full control and hidden control view.
When the controls are hidden the display expands to occupy the full window.

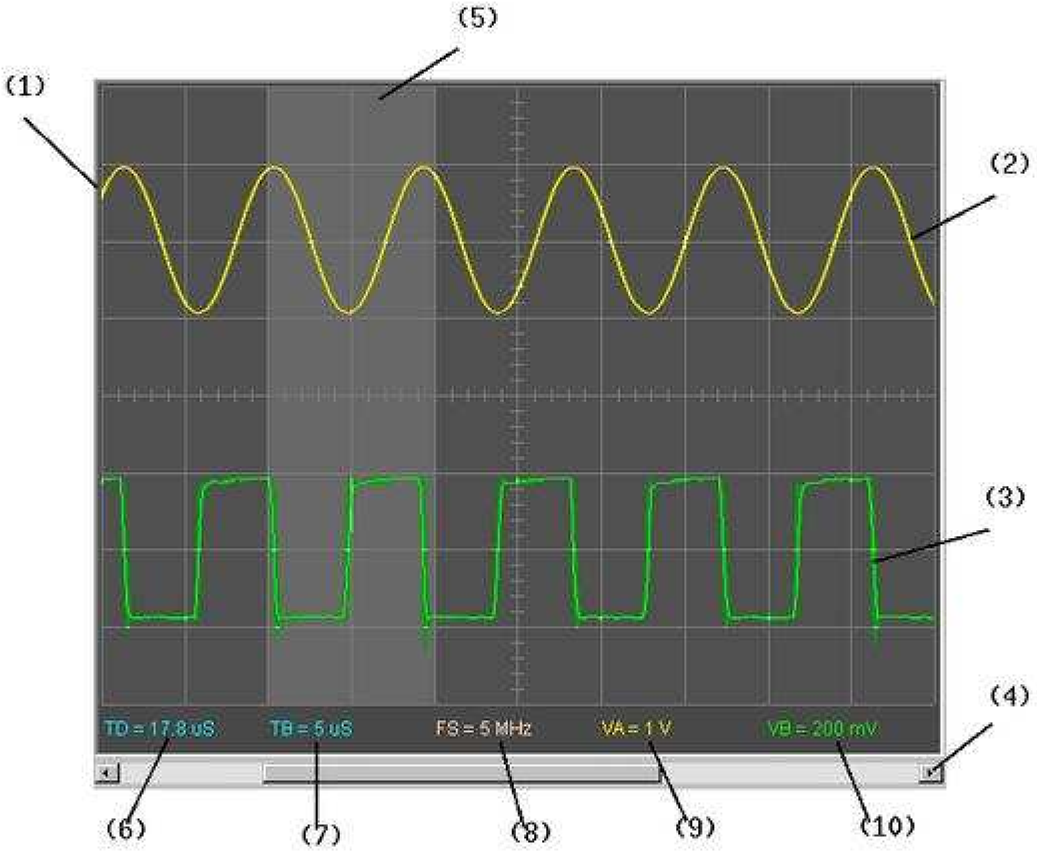
GRID enables the display graticule.

For regular oscilloscope operation it will normally be enabled. When working with display cursors or when inspecting waveforms in fine details it is often convenient to disable it.

SCOPE DISPLAY

The display shows live or replayed waveforms, spectra and logic

This display shown here is probably the most familiar; a dual trace digital oscilloscope.



TD = 17.8 uS TB = 5 uS FS = 5 MHz VA = 1 V VB = 200 mV

(1) Display Graticule: all waveforms are displayed on the 8 x 10 graticule.

(2) Channel A Trace: Channel A trace drawn in yellow.

(3) Channel B Trace: Channel B trace drawn in green.

(4) Waveform Offset: used to scroll both waveforms left and right.

(5) Zoom Band: indicates where the zoom timebase will operate (when enabled).

(6) Trigger Delay: time of the left edge of the display referenced to the trigger.

(7) Timebase: the prevailing timebase setting (time per division).

(8) Sample Rate: display sample rate (note this may be different to the capture rate).

(9) V/Div Channel A: vertical scaling channel A (volts per division).

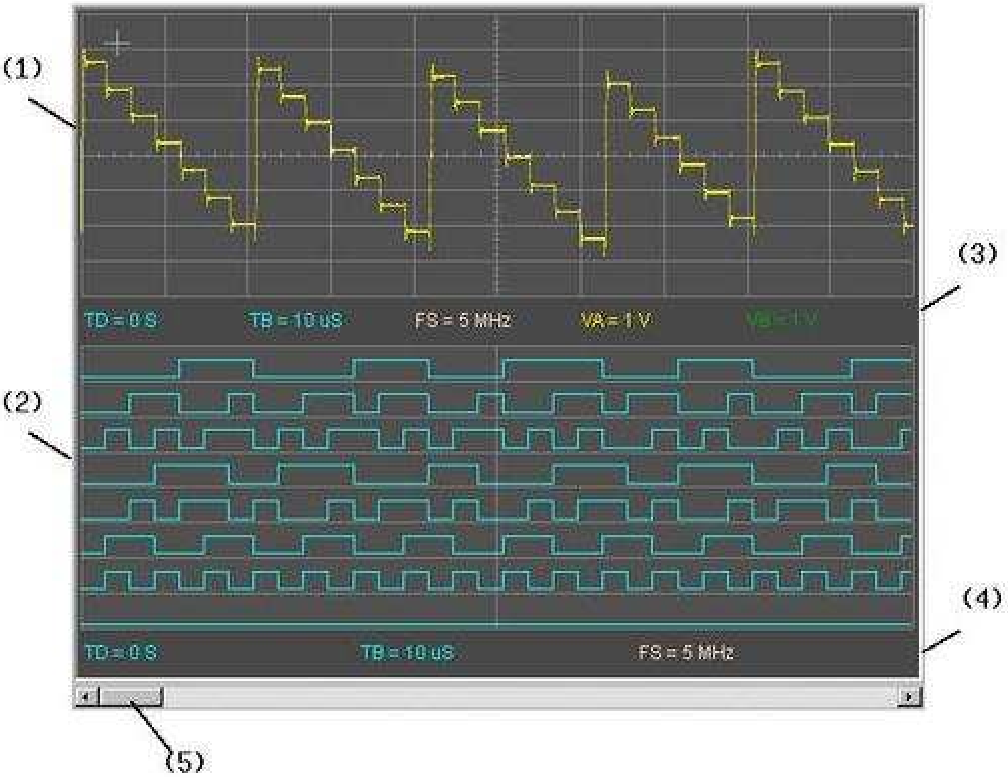
(10) V/Div Channel B: vertical scaling channel B (volts per division).

By default the **timebase zoom** is set such that the display shows 1/2 of the total capture buffer (ie, the waveform offset slider **(4)** is 1/2 of the display width).

This makes it easy view any feature of the waveform on the display using **Waveform Offset**.

Mixed Signal Display

When used to capture analog and logic signals the default is a split analog and logic display.



TD = 0 S TB = 10 uS FS = 5 MHz VA = 1 V VB = 1 V

TD = 0 S TB = 10 uS FS = 5 MHz

(1) Analog Display: analog trace displayed on the 8 x 10 graticule.

(2) Logic Display: 8 logic traces displayed concurrently with the analog trace.

(3) Analog Info: same information set as the dual channel analog display.

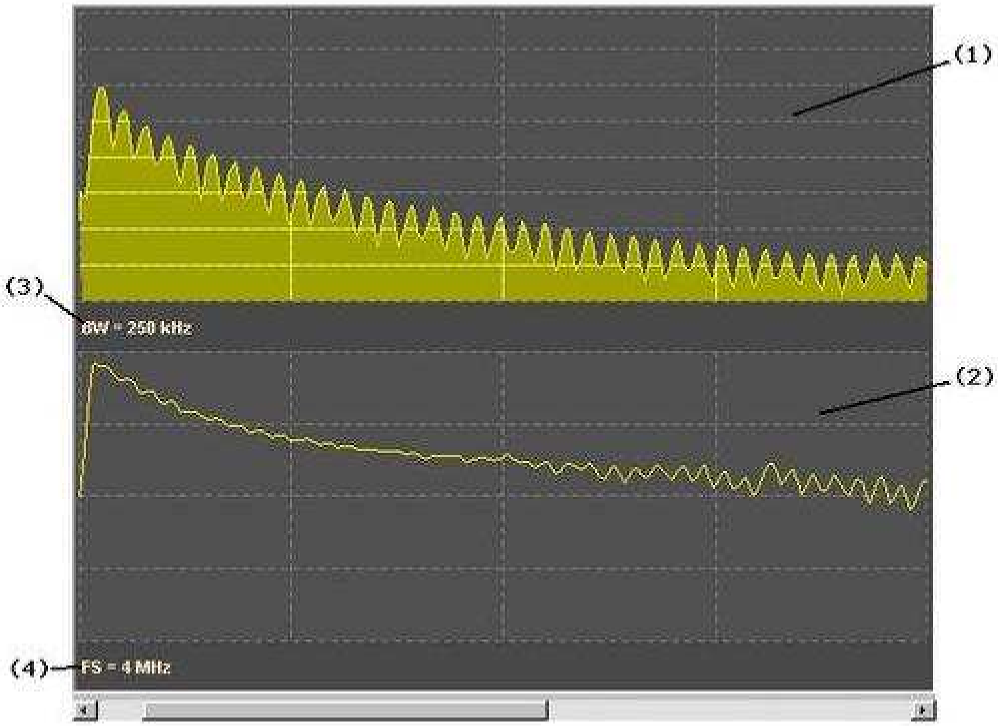
(4) Logic Info: a sub-set of information pertaining to the logic display.

(5) Waveform Offset: used to scroll both analog/logic displays left and right.

If viewing only logic or analog captured in mixed mode, the split display can be turned off to show just logic (or analog) using the full display. The timebase zoom in mixed and logic VIs are usually set more "zoomed in" than in other modes because usually you're looking for timing detail rather than overall analog waveshape. Of course you can adjust the timebase zoom at any time to suit your preference.

Spectrum Display

Several spectrum display formats are available. This one shows magnitude and phase.



(1) Magnitude Spectrum: waveform magnitude (amplitide) spectrum in dB (10 dB/Div).

(2) Phase Spectrum: waveform phase spectrum (90 Deg/Div).

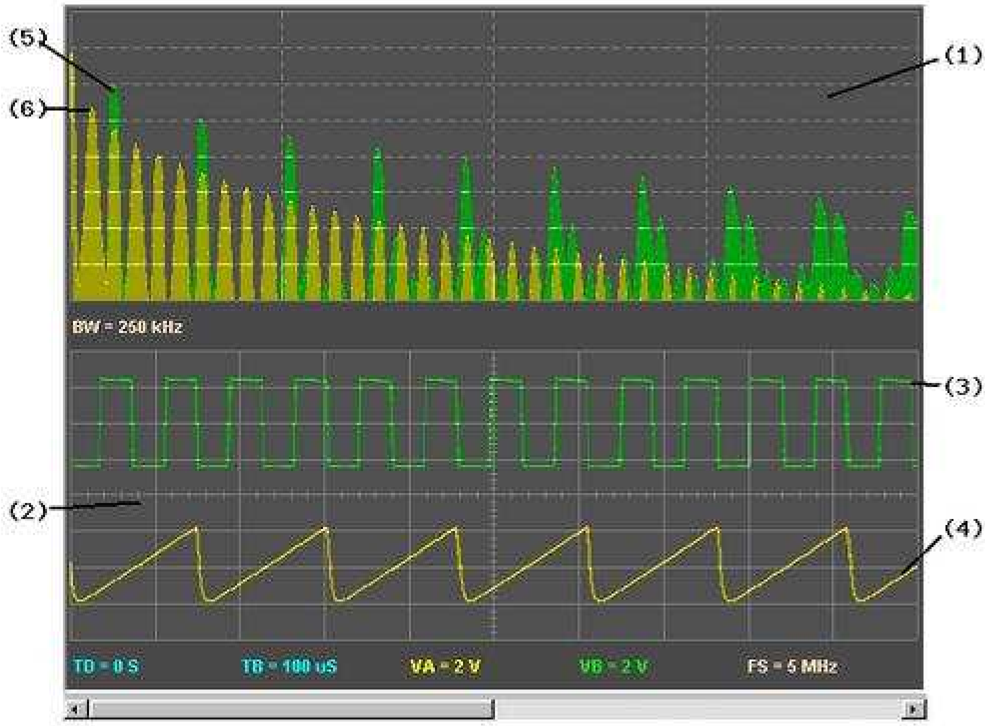
(3) Spectrum Bandwidth: Full spectrum bandwidth (250 kHz in this case).

(4) Sample Rate: Capture sample rate (Hz).

The spectrum analyzer displays are tightly integrated with the main DSO controls; sample rate, spectrum analyzer bandwidth and reference levels are all set using standard DSO controls (ie, timebase and channel V/Div). As with other displays it is possible to turn off the split display to show just the magnitude spectrum using the full display.

Split Waveform/Spectrum Display

Frequently it is desirable to see the waveform a spectrum together.



(1) Magnitude Spectrum: waveform magnitude (amplitide) spectrum in dB (10 dB/Div).

(2) Waveform Display: waveform time display (V/Div per channel).

(3) Channel B: Channel B waveform display (square wave).

(4) Channel A: Channel A waveform display (rounded sawtooth).

(5) Channel B: Channel B magnitude spectrum (first harmonic).

(6) Channel A: Channel A magnitude spectrum (first harminic).

This example shows both channels' waveform and magnitude spectra simultaneously.

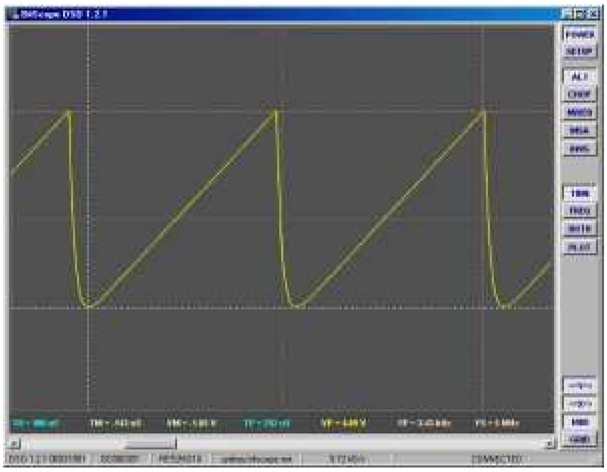
The display mode is Dot/Vector (ie, high resolution) and all the DSO controls remain available allowing you to (for example) zoom in on a region of waveform an look at the spectrum of the region simultanouesly or enable cursors in both time and frequency domains to make concurrent measurements in the two different views of the signals.

This was done on **SYDNEY** so you can try this yourself, even if you don't have a BitScope handy.

DISPLAY CURSORS

Cursors allow measurements of time, frequency, voltage and level

Cursor operation is the same from a user-interface point of view in all cases.



The screenshot to the right shows the analog waveform display with all waveform cursors enabled and the graticule disabled.

Cursor Types

There are two movable cursors per axis:

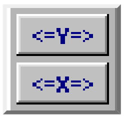
- **POINT** the measurement cursor (RED).
- **MARK** the reference cursor (WHITE).

Both appear as dashed horizontal or vertical lines on the display.

The POINT cursor is the primary cursor.

It may be used on its own to measure absolute values in units of the display axis on which it appears.

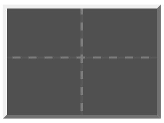
The MARK cursor is the reference cursor.



If moved from its default (zero) position it marks a reference value.
The POINT cursor then uses this to report its measurements *relative* to the MARK.

Cursors for each axis are enabled via the **<=X=>** and **<=Y=>** buttons.

The Origin Cursor



A third (immovable) cursor (GREY) appears in some displays to identify the "origin" (eg, a trigger point or a channel's ground).

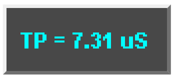
The POINT and MARK cursors default position is at the origin.
That is, the origin is where you'll find them located when first you enable them.

Cursor Movement

To move a cursor, click on it (with the left mouse button) and drag it to the position required.

If both cursors are at the same location, the POINT cursor will be picked up first.
For this reason it is generally a good idea to locate the POINT cursor to the required position first.

Cursor Values

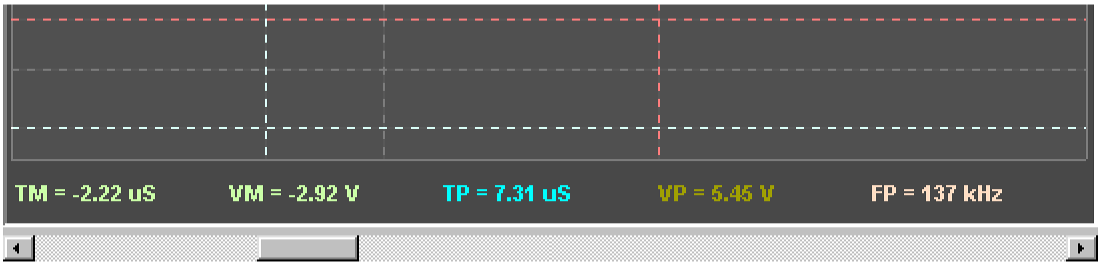


Usually two cursor values are reported at the bottom of the display; the absolution value (ie, referenced to the origin) of the MARK cursor and the relative value (ie, referenced to the MARK) of the POINT cursor.

Because the MARK defaults to the origin, moving just the POINT gives you an absolute value referenced to the origin. If you then move the MARK, you can make measurements between two arbitrary points on a waveform, logic trace or spectrum display.

Cursor Display

Fig **[1]** shows a cut-out of the main display with both **<=X=>** and **<=Y=>** cursors enabled.



TM = -2.22 uS VM = -2.92 V TP = 7.31 uS VP = 5.45 V FP = 137 kHz

Fig [1] - Cursors Display

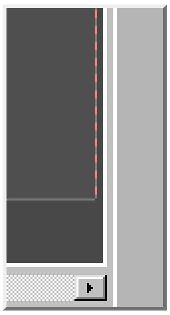
The feint grey cross-hairs are the X and Y axis origin cursors. The vertical cursor identifies the trigger point. The horizontal cursor indicates zero voltage (ie, ground) on the selected channel.

The vertical red and white cursors are the time and/or period measurement cursors. Vertical cursors on spectrum displays are similar but measure frequencies and/or bandwidths in Hz.

The horizontal red and white cursors are voltage and offset measurement cursors. Horizontal cursors on spectrum displays are similar but measure spectrum magnitudes in dB.

The values measured by the various cursors are show below the main display.

Off Screen Cursor Location



If a cursor's value is such that it would be positioned "off screen" it will instead be shown "pinned" to the left or right (or top or bottom) of the display.

Consequently it is possible to see which way to adjust the horizontal or vertical offset to locate the cursor onto the display. It also means the cursor is always available to be "picked up" with the mouse (ie, you can't "loose" cursors).

Note "can't loose cursors" does not mean they are not sometimes hard to see.

The display GRID can obscure cursors somewhat (we recommend you turn off the GRID if using cursors) as can the waveform or spectrum itself, or another cursor. In the latter case, a quick drag of the mouse will determine whether one cursor is sitting atop another.

Virtual Instruments

While the DSO implements multiple **virtual instruments** there is only one set of cursors for each type of measurement (ie, time, frequency, voltage and level) regardless of the selected VI.

This means you can set up cursor positions referenced to (say) a logic transition in the logic analyzer and (assuming you're using the same trigger condition) you can switch to the SCOPE mode digital oscilloscope and see where in the analog waveform those transitions are located.

Of course MIXED mode provides similar capabilities all on the one display but you may prefer the SCOPE's more powerful waveform analysis tools or use the SCOPE to establish a "zoomed in" view of the waveform. In both cases however, the cursor positions will be reported and displayed with the same values making these cross-VI measurements possible.

Time and Frequency Cursors

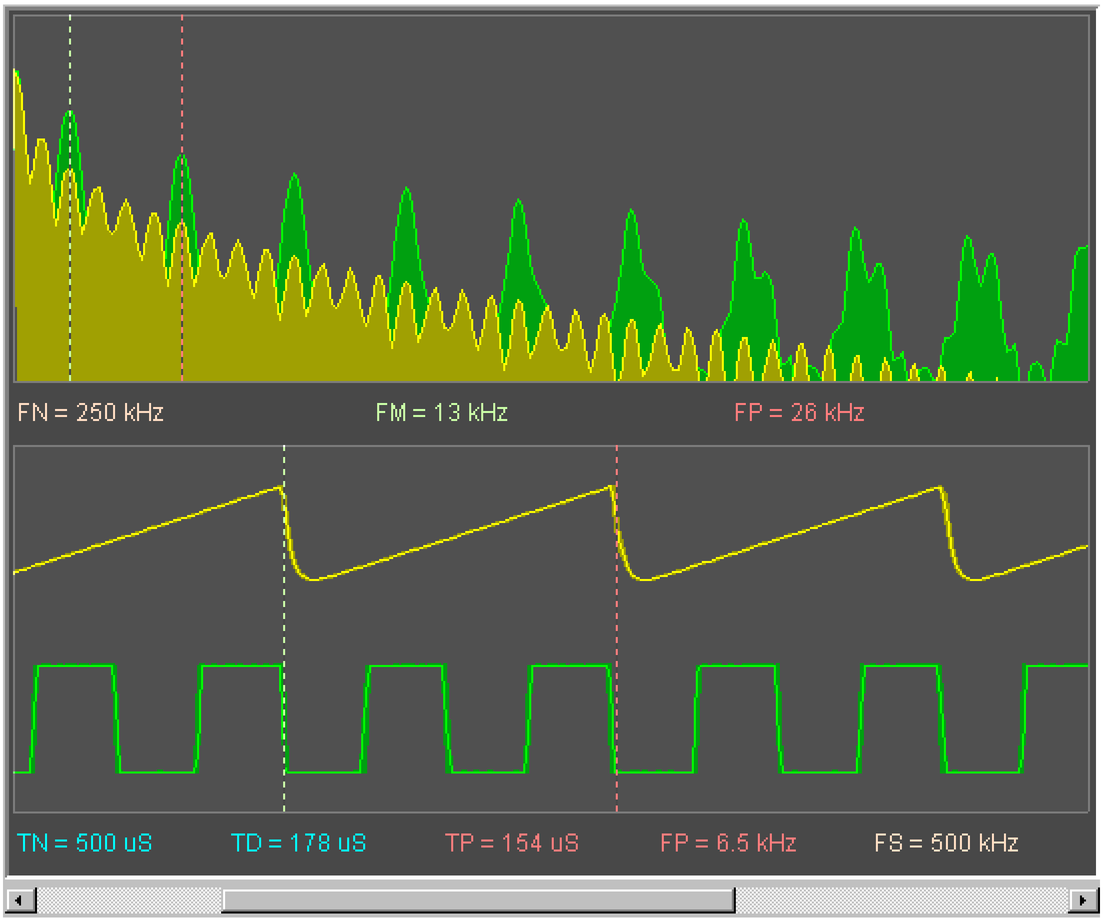
<=X=>

The DSO supports TIME and FREQUENCY cursors for precise measurement of time periods, pulse width, delays, frequencies and bandwidths.

These cursors are also useful to indicate fixed time and frequency points when adjusting waveform or frequency offsets, applying a post-trigger delay or adjusting timebases or timescales.

They are enabled via the **<=X=>** button which also enables the display of **cursor variables** so measurements can be made.

Below is a screenshot showing both cursors on features of the SYDNEY Bitscope waveforms.



FN = 250 kHz FM = 13 kHz FP = 26 kHz

TN = 500 uS TD = 178 uS TP = 154 uS FP = 6.5 kHz FS = 500 kHz

Fig [2] - SYDNEY BitScope with Time and Frequency cursors enabled.

In this example the time cursors are positioned to report the waveform period on channel A (TP=154 uS) and the associated frequency (FP=6.5 kHz).

The frequency cursors are positioned on the first harmonic (fundamental) of the waveform on channel B (FM=13 kHz) and the third harmonic (39 kHz). The frequency difference between the first and third harmonics is 26 kHz (FP=26 kHz).

These measurements in both time and frequency domains are consistent (the waveform on channel B is precisely twice the frequency of that on channel A).

Some other statistics to note are the display sample rate (FS=500 kHz), the total spectrum bandwidth (FN=250 kHz) and the time span of the waveform display (TN=500 uS). Also, the left edge of the time display is 178 uS after the trigger point.

In normal usage, if you are interested to measure a single time or frequency value (referenced to 0 Hz or the trigger point respectively) simply move the Point Cursor (RED) to the position you wish to measure. If you want to measure the **difference** between two times or frequencies move the Point Cursor to the right point (first) and the Mark Cursor (WHITE) to the left point (second).

Once located to a point in time or a frequency, each cursor will remain there regardless of the timebase, time scale or DSO operating mode. This can be very useful if (for example) you locate to a particular position in a logic sequence (in LOGIC mode) and switch to the scope (ALT or CHOP mode) to see the analog signal in fine detail.

Voltage and Level Cursors

The DSO also provides VOLTAGE (waveform displays) and LEVEL (spectrum displays) cursors for precise measurement of voltages, voltage differences and spectrum levels. As with the **<=X=>** cursors, there are two **<=Y=>** cursors **POINT** and **MARK** which are manipulated in the same way.

The following screen shot shows the waveform **<=Y=>** cursors in use:

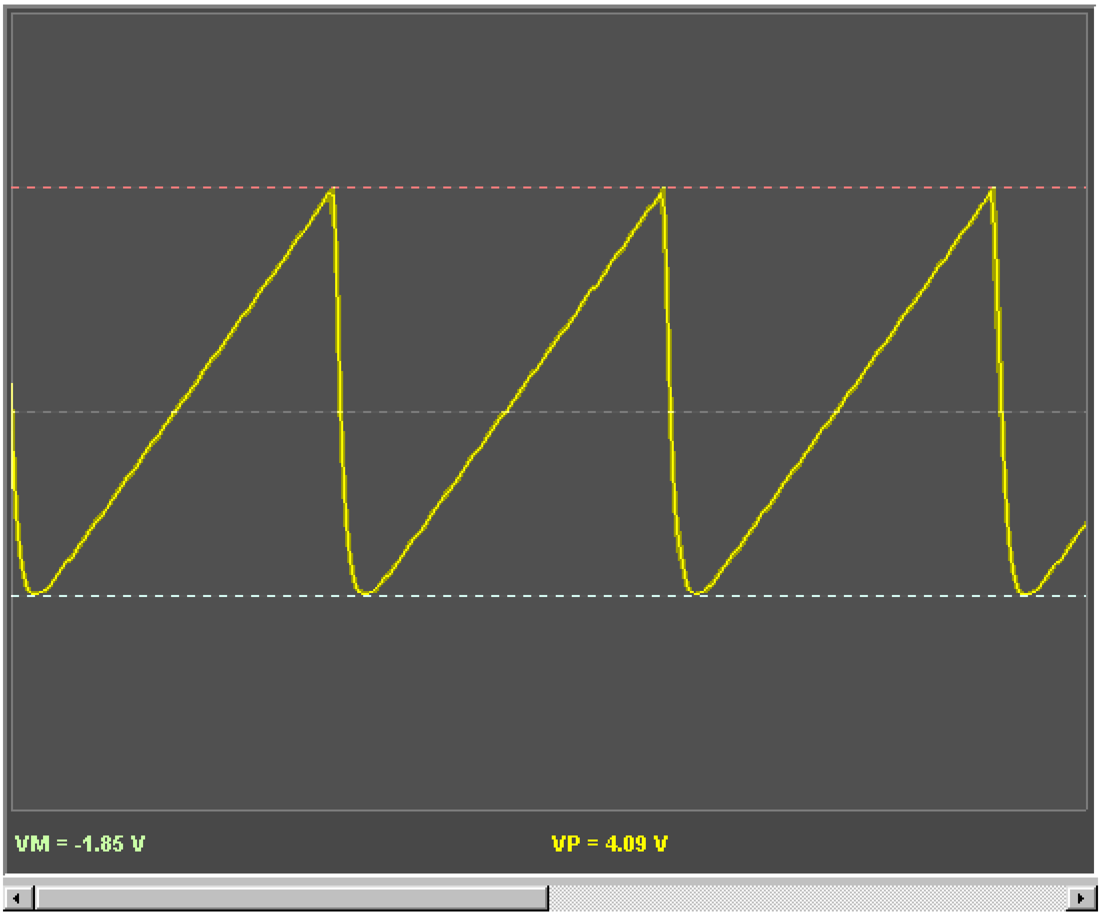


Fig [3] - SYDNEY BitScope with Voltage cursors enabled.

In this example (also from SYDNEY) the analog waveform on Channel A is 4.09V peak-to-peak starting from a negative voltage of -1.85V. The grey line in the middle is zero volts (ground).

It is important to understand that the **<=Y=>** cursors are associated with a channel and not the display itself. This is indicated with an underline below the channel's name.

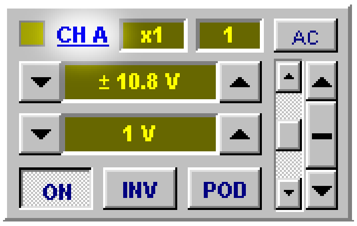


Fig [4] - Channel Selection (CH A)

By default the selected channel is CH A but simply clicking on another channel name (eg, CH B) selects that channel instead.

When a different channel is selected, the cursors may move to another position on the display if the newly selected channel's vertical scaling (V/Div) or offset is different from the previously selected channel. That is, the cursors retain the same voltage (or level) but the position on the display at which this voltage or level is located may be different (so the cursors move relative to the display).

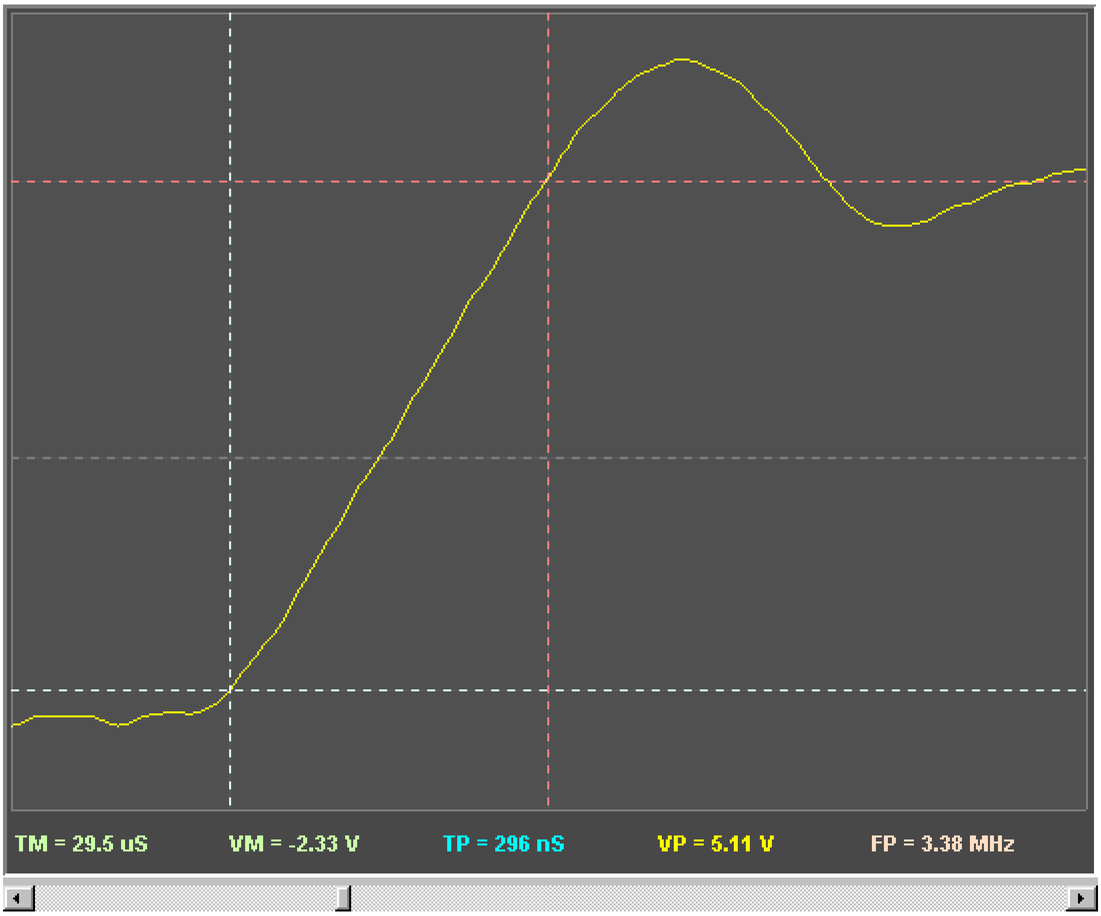
Because the cursors report measured voltage on a channel it is possible to make measurements on one channel and switch to the other channel to make measurements relative to the first channel.

Similarly, voltage difference measurements can be made very precisely even if quite large by increasing the vertical sensitivity and adjusting the offset (if necessary) to place the MARK cursor and then again adjusting these controls appropriately to place the POINT cursor even if it is not possible to see both on the display at the same time for a given vertical scaling.

Slew Rate Example

A common use for combined cursor measurements is rise time or slew rate measurements.

Using the DSO's cursors together such measurements are made easy:



TM = 29.5 uS VM = -2.33 V TP = 296 nS VP = 5.11 V FP = 3.38 MHz

Fig [5] - Slew Rate Example.

In this example the waveform on channel A rises from -2.33V to 2.78V (a total of 5.11V) with the edge starting 29.5 uS after the trigger point rising for 296nS; a slew rate of 17.3 V per millsecond.

Mixed Signal Timing Example

Time cursors are extensively used with the mixed mode and logic analyser displays.

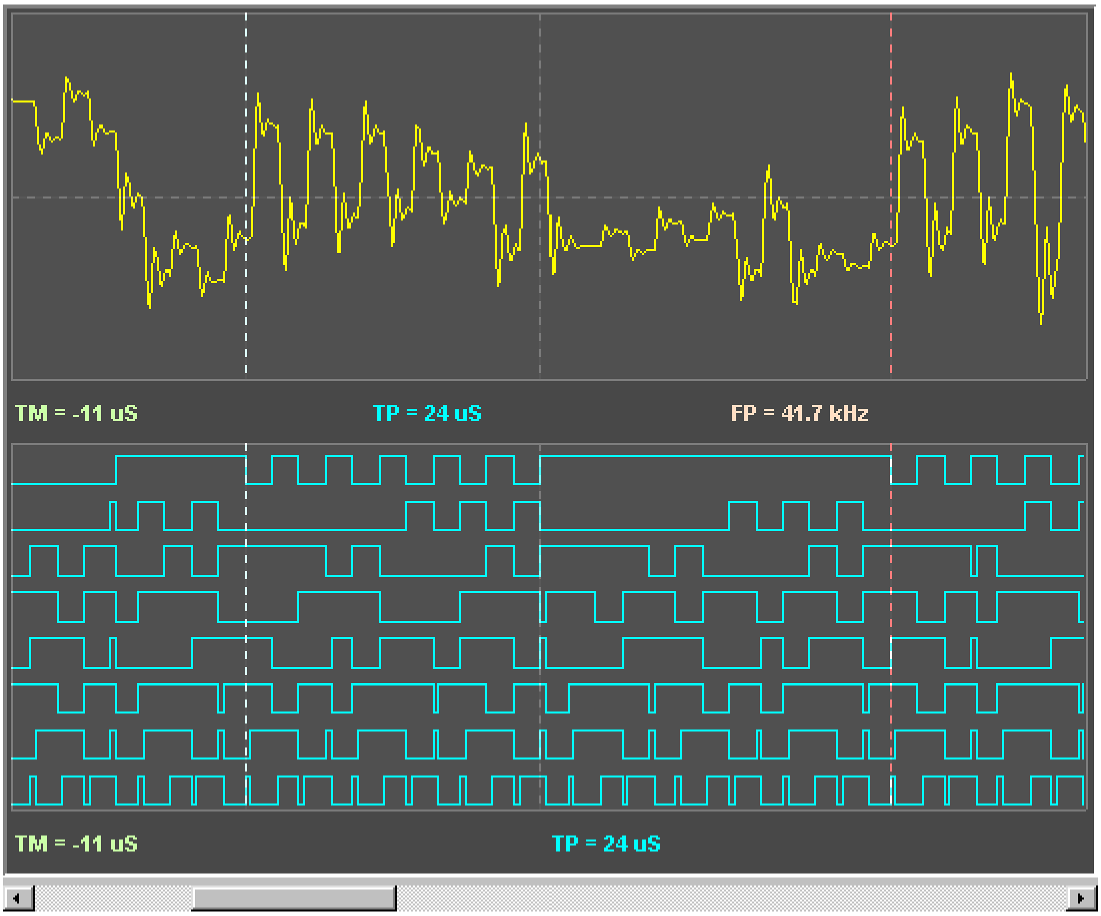


Fig [6] - Logic Timing example.

The trigger is shown in the center of the display.

The MARK cursor is placed at the start of the burst of edges on Bit 7 before the trigger.

The POINT cursor placed on the start of the next burst after the trigger, also on Bit 7.

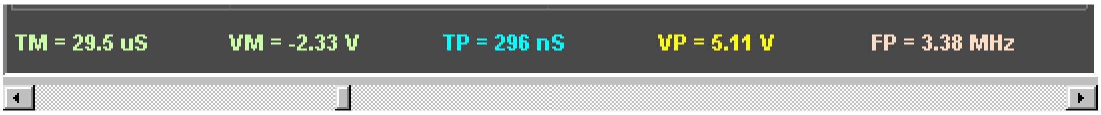
Timing cursors are locked together between the analog and logic displays so timing measurements may be made between analog and related logic events. In this case Bit 7 is the most significant bit of a complex analog waveform produced as the D/A convertion of the 8 logic signals.

The first burst (at the MARK) appears 11 uS before the trigger and the second appears 24 uS after the MARK (13 uS after the trigger). The frequency implied by the MARK to POINT period is 41.7kHz.

SCOPE VARIABLES

Scope variables report operating parameters and capture statistics

Scope variables are shown below the main display:



TM = 29.5 uS VM = -2.33 V TP = 296 nS VP = 5.11 V FP = 3.38 MHz

The set of variables shown depend on the display type and whether features such as **cursors** are enabled. In the screenshot above, the variables associated with time and voltages cursors are shown because both cursors were selected when it was taken.

Variables are identified by a two character name and are typically colour coded on the display:

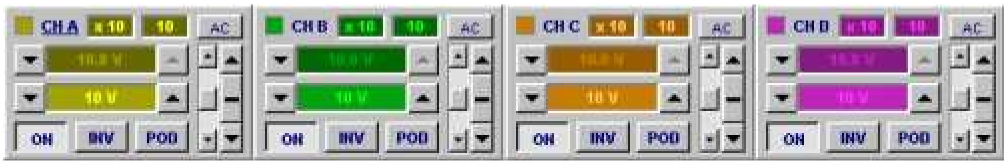
ID	**NAME**	**DESCRIPTION**
TB	Timebase	Time per division (Graticule Enabled) or time per display (Graticule Disabled).
BW	Bandwidth	Spectrum display bandwidth or Nyquist frequency (aka the highest displayed frequency).
FS	Sample Rate	Display or capture sample rate. Which it reports depends on the data mode (ie, it reports the effective sample rate of the waveform).
TD	Time Delay	Pre/Post-trigger time delay (non-zero if pre-trigger, delay or time offset has been set).
VA	Voltage Scale	Volts per division (Graticule Enabled) or Volts per display (Graticule Disabled) for Channel A. Other channels (VB, VC and VD) similar.
VM	Voltage (Mark)	Voltage at the Mark Cursor (Blue) on the selected channel. Reported in the colour of the selected channel (eg, Yellow for CH A).
VP	Voltage (Point)	Voltage at the Point Cursor (Red) relative to the Mark. Reported in the colour of the selected channel (eg, Yellow for CH A).
TM	Time Period (Mark)	The time offset from the trigger point to the time mark cursor.
TP	Time Period (Point)	The time difference between the two time cursors.
FM	Frequency (Mark)	The value of the Mark frequency cursor. Appears in spectrum displays only.
FP	Frequency (Point)	The difference (aka bandwidth) between the Point frequency cursor and the Mark frequency cursor. Appears in spectrum displays only.
MM	Magnitude (Mark)	The value of the Mark level cursor (in dB). Appears in spectrum displays only.
MP	Magnitude (Point)	The difference between the Magnitude Point and Mark cursors. Appears in spectrum displays only.

An important distinction between these variables and DSO parameters that control them is that variables on the display always associate with the waveform, spectrum or logic data currently displayed.

This is in contrast to some DSO parameters (eg, **timebase**) which may be changed to new values which do not correspond to the displayed waveforms until a new capture is performed.

CHANNEL CONTROLS

Channel controls determine capture voltage range, scale and offset



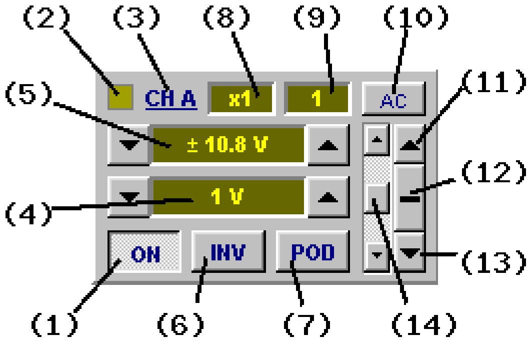
An array of control panels, one for each analog channel, is positioned below the main display.

The number of control panels that appear depends on the type of BitScope.

The figure above shows the configuration for **BS442N**.

And to the right, one channel on **BS310**.

Other models may be slightly different.



(1) **ON** enables channel capture and display.

(2) **Enable LED** shown dimly in the color of the channel when enabled.

Flashes brightly when a capture occurs.

Flashes in **RED** if overloaded.

(3) **Channel Select** identifies the channel and whether it is selected. Select a channel by clicking its name. The selected channel is indicated with an underline (as shown).

(4) **Display Scale** selects the volts per division at which to display the waveform.

(5) **Input Range** selects the analog input range used for capture. Usually automatically assigned.

(6) **Signal Invert** click to invert the waveform.

(7) **Input Source** selects the channel source; BNC input or POD connector.

(8) **Probe Scale** rescales the display for use with x10, x100 or x1000 oscilloscope probes.

(9) **Prescaler** select the analog prescaler (on BitScope models that support it).

(10) **Coupling** selects AC input coupling (DC coupling is the default).

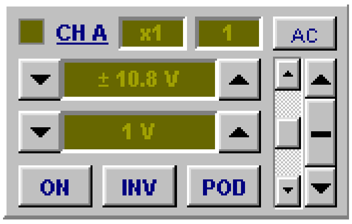
(11)..(14) **Offset Control** move the waveform up and down on the display.

(12) **Offset Calibration** right-click the zero position button to calibrate offsets.

Channel Enable and Waveform Capture

(1) (2)

Each analog channel may be individually enabled or disabled.



When disabled the channel control panel's widgets are shown in feint colors indicating their "turned off" status.

Disabling a channel deselects capture to that channel but it may still be selected as a **trigger source**.

It is usually a good idea to disable a channel if you don't need it because doing so may free more capture buffer to be used for the other enabled channels (depending on BitScope model).

When disabled most of the channel's controls can still be modified. Any changes made will be applied next time the channel is enabled.

[1] Channel A in disabled state

Range and Scale

(5) (4)

Analog waveform capture and display depends on the selected input range and vertical scale.

Vertical **SCALE** adjusts the vertical height of the waveform.

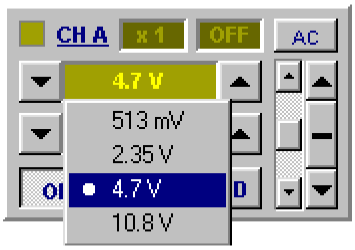
It will be familiar if you have used an oscilloscope before.

Defined in "Volts per Division" (V/Div) it changes in a 1, 2, 5, 10, 20 ... sequence each time the UP or DOWN buttons are clicked.

Input **RANGE** adjusts the gain applied to the analog signal.

This may be less familiar but its assignment is very important.

Both parameters may be changed by clicking the UP/DOWN buttons on either side of the parameter display.



[2] Range Menu Selection

They may also be selected by right-clicking the parameter itself to pop-up a menu of available values (as shown in FIG **[2]** for the range parameter.

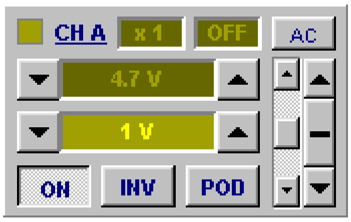
Autorange mode

Usually the range is assigned *automatically* based on the scale.

This is called *Autorange* as shown in Fig **[3]**.

It is indicated by the scale highlight.

The range is automatically selected to optimize the digital resolution when the analog signal is captured which for most purposes is the preferred mode of operation.



[3] Autorange mode

Autorange is the default configuration for the analog channels.

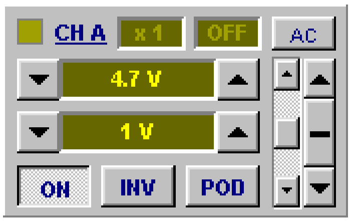
Manual mode

There may be occasions when you need to explicity choose which input range you want to use.

For example, you may need to choose the range based not on its attentuation but on its bandwidth or some other criteria.

Clicking the scale parameter or simply selecting a different range selects *Manual* mode as shown in Fig **[4]**.

In this mode neither range nor scale are shown with highlight and both parameters are independently adjustable.



[4] Manual mode

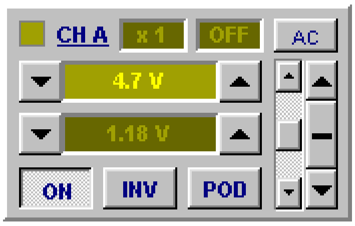
Normalize mode

Sometimes it can be difficult to "find" a signal on the display. For example, this can occur when a signal has a large DC offset which moves the waveform beyond top or bottom of the display.

In this case clicking the range parameter activates *Normalize* mode as shown in Fig **[5]** which can be very helpful.

This mode assigns the scale according to the selected range to normalize the display to the full scale digitized waveform.

As shown here this can produce some unusual scale values (1.18 V/Div in this case) but it means that the signal, even if it's overloading the channel, can be seen on the display.



Of course in *Normalize* mode you can still use **voltage cursors** instead of the graticule to make measurements.

[5] Normalize mode

Normalize mode is also needed for another purpose; **offset calibration**.

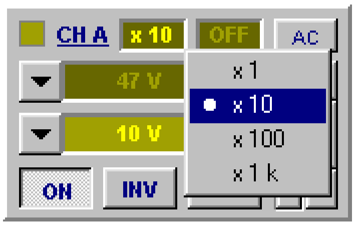
Probe Scaling (8)

The BNC inputs may be scaled for the type of probe connected (accounting for its attenuation).

The set of probe attenuations are x1, x10, x100 and x1000.

These may be changed by left clicking the value (to toggle between two values) or right clicking (to select a value directly).

Fig **[6]** shows Channel A with 1V/Div scale and 4.7V range selected by with x10 probe attenutation selected.



[6] Probe Scaling

This results in the channel having a 47V input range with 10/Div (ie, at the probe tip).

Using an attenuating probes you can extend BitScope's maximum measurable voltage range.

For example a switchable x1/x10 probe in x10 mode allows measurement of up to +/- 108V (via the 10.8V range).

Likewise a x100 probe allows measurement to +/- 1000V.

In practice we highly recommend the use of a fully isolated differential probe such as the one pictured in Fig **[7]** when working at very high voltages with BitScope.

[7] Isolated Differential Probe

We also recommend you use an isolated BitScope such as our Network BitScope models **BS310N** or **BS442N** or if using a USB BitScope that you reference it to the PC's ground or use an isolated PC (like a notebook).

Analog Prescale (9)

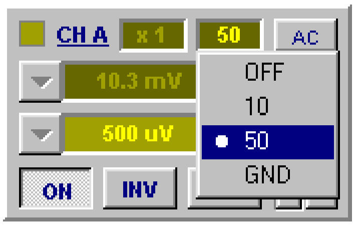
The BNC inputs may be prescaled for measurement of very low level signals (on some models).

The set of prescale values available depends on the BitScope model used.

In the case of **BS310** it is OFF (x1), x10, x50 and GND.

As with **probe scaling** the analog prescale may be changed by left clicking the value (to toggle between two values) or right clicking (to select a value directly).

Fig **[8]** shows Channel B with the 513mV input range and the x50 prescaler selected which results in the channel having a 10.3mV input range (ie, *very small*) with very high sensitivity.



[8] Analog Prescale

The scale is set to 500uV/Div in this example to demonstrate just how sensitive the analog channels can be with prescalers.

However, if you are to make effective use of such a high input gain you need to ensure the waveform you're measuring is (in this example) less than 20.6mV peak to peak.

You may also need to apply an **offset** to locate a small signal sitting on a DC voltage to ground and be aware that at these very low levels *all signals are very "noisy"*.

(10)...(14)

Vertical Offset and Coupling

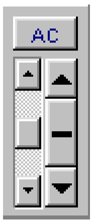
Locating the position of the waveform on the display can be done in different ways.

In general the idea is to locate the waveform to the centre of the display and maximize its size for measurement purposes. Another use for the position controls is to vertically separate multiple channels when displaying them simultaneously.

In all these cases the offset and coupling controls shown in Fig **[9]** are used to adjust the waveforms' vertical position. How the offset is actually applied depends on which model BitScope and which channel input (BNC or POD) you're using.

There are two types of offset applied when you adjust the vertical position controls:

- **Digital Offset** applied after A/D conversion.
- **Analog Offset** applied before A/D conversion.



[9] Offset

In general you do not need to be concerned with the difference between them except to understand that high resolution measurement of small scale signals sitting on top of large DC offsets (or large signals) require Analog Offset (and/or AC Coupling) to locate the signal to the centre of the display to facilitate the use of **analog prescalers** or sensitive **input ranges**.

(12)

Offsets and Calibration

All internal analog channel offsets in BitScope are factory calibrated.

This means that when the inputs are ground referenced (ie, connected to ground) and the offset control is reset, the trace will appear in the center of display (ie, at exactly zero volts).

However there are circumstances where you may wish to make changes to this offset calibration; temporarily to eliminate offsets appearing in the signal under test under specific circumstances and permanently to recalibrate the channel if temperature or the passage of time make it necessary.

Offset Assignment

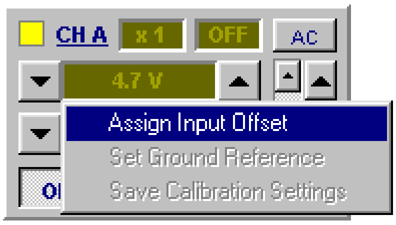
Sometimes a DC offset may appear in a signal under test which you want to ignore.

Alternatively you may find that for precise measurement you wish to finesse the calibration of a given range or prescale.

Offset calibration is available via the menu shown in Fig **[10]** that appears upon right-clicking the zero offset button **(12)**.

For a given source, range and prescale and the input referenced, offset controls can relocate the trace to the center of the display.

With the trace centered, selecting *Assign Input Offset* recalibrates the channel offset for that channel setup.



[10] Input Offset Assignment

That is, it assigns a temporary calibration offset to the channel which ensures that the reference signal appears exactly at the center of the display.

All subsequent measurements will be made referenced to the assigned offset not zero volts.

Offset Calibration

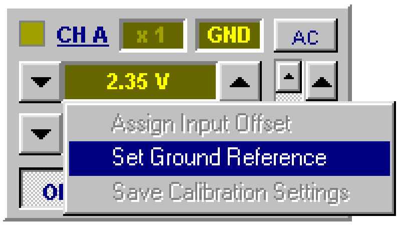
If offsets appear on a channel on a particular source, range or prescale when the input is properly *ground referenced*, the internal channel offsets may require persistent recalibration.

This is achieved the same way as *offset assignment* described above but there is a specific **procedure** that must then be followed to calibrate all the channel offsets correctly.

It is important to understand that an internal offset can appear in each possible signal path through the analog channel circuitry.

That is, a different calibration parameter may be required for each source, range and prescale value. In some BitScopes this adds up to quite a few parameters (for example there are **20** possible parameters for each channel in **BS310**).

In practice the number of *different* parameters is much smaller (5 per channel in **BS310**) so in relatively few steps it is possible to recalibrate *all* channel offsets.



[11] Set Ground Reference

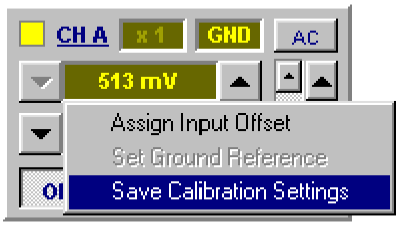
The first step in all cases is assignment of the channel's global ground reference.

When a specific channel source, range and prescale is selected, the channel's global ground reference may be assigned as shown in Fig **[11]**.

Next a small set of input offsets for other source, range and prescale values are assigned. At this point they may be saved to the BitScope as shown in Fig **[12]**.

When this is done *all* the channel calibration parameters are recalculated based on the assigned input offsets and saved to *flash memory* to be reused automatically in future.

The set of channel settings for which the input offset must be (re)assigned to enable this complete recalibration of the channel depends on the BitScope model concerned.



[12] Save Calibration Settings

Offset Calibration Procedures

The tables below enumerate the offset **calibration** steps per channel for each BitScope model.

All inputs must be ground referenced and channels **normalized** before attempting calibration.

BitScope Model 310 Channel Offset Calibration Procedure

Step	Source	Range	Action
STEP 1	GND	2.35 V	Set Ground Reference
STEP 2	BNC	2.35 V	Assign Input Offset (Prescale OFF)
STEP 3	BNC x10	235 mV	Assign Input Offset
STEP 4	BNC x50	47 mV	Assign Input Offset
STEP 5	GND	513 mV	Assign Input Offset
STEP 6	Any	Any	Save Calibration Settings

Note: it is recommended that both channels be calibrated simultaneously.

BitScope Pocket Analyzer Channel Offset Calibration Procedure

Step	Source	Range	Action
STEP 1	POD	1.22 V	Set Ground Reference
STEP 2	BNC	2.35 V	Assign Input Offset (Prescale OFF)
STEP 3	BNC x10	235 mV	Assign Input Offset
STEP 4	POD	267 mV	Assign Input Offset
STEP 5	Any	Any	Save Calibration Settings

Note: only channel A needs to be calibrated.

BitScope Model 442 Channel Offset Calibration Procedure

Step	Source	Range	Action
STEP 1	GND	2.35 V	Set Ground Reference
STEP 2	BNC	2.35 V	Assign Input Offset (Prescale OFF)
STEP 3	BNC x10	235 mV	Assign Input Offset
STEP 4	GND	513 mV	Assign Input Offset
STEP 5	Any	Any	Save Calibration Settings

Note: each channel may be calibrated independently of the others.

BitScope Model 300 Channel Offset Calibration Procedure

Step	Source	Range	Action
STEP 1	POD	1.2 V	Set Ground Reference
STEP 2	BNC	1.2 V	Assign Input Offset
STEP 3	POD	262 mV	Assign Input Offset
STEP 4	Any	Any	Save Calibration Settings

Note: it is recommended that both channels be calibrated simultaneously.

BitScope Model 440 Channel Offset Calibration Procedure

Step	Source	Range	Action
STEP 1	GND	600 mV	Set Ground Reference
STEP 2	BNC	600 mV	Assign Input Offset
STEP 3	ALT	3.39 V	Assign Input Offset
STEP 4	GND	130 mV	Assign Input Offset
STEP 5	Any	Any	Save Calibration Settings

Note: each channel may be calibrated independently of the others.

BitScope Model 220 Channel Offset Calibration Procedure

Step	Source	Range	Action
STEP 1	POD	2.92 V	Set Ground Reference
STEP 2	BNC	600 mV	Assign Input Offset
STEP 3	POD	632 mV	Assign Input Offset
STEP 4	Any	Any	Save Calibration Settings

Note: it is recommended that both channels be calibrated simultaneously.

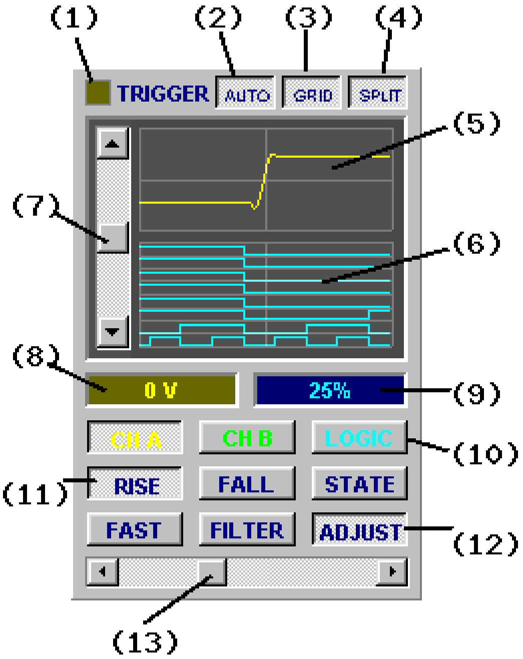
TRIGGER CONTROL

The trigger control sets conditions for waveform and logic capture

As with traditional oscilloscopes, the trigger establishes a time reference for captured waveforms.

However because BitScope is a mixed signal system, triggering is a little more sophisticated than most stand-alone oscilloscopes. In particular you can trigger the capture of analog waveforms on logic conditions and vice-versa as well as apply digital triggers to analog signals.

The trigger panel is where you set up triggers.



It comprises a **Trigger Display** showing the analog and/or logic signals as they appear at the trigger point and a set of trigger control buttons and parameters to establish trigger conditions.

(1) **Trigger LED** lights **RED** when waiting for a trigger switching to the color of the trigger channel (yellow, green or blue) when the trigger fires.

(2) **AUTO** enables automatic trigger which fires after a short delay if a trigger event is not seen.

Auto trigger is most useful in REPEAT mode when you're not sure what the signal is.

(3) **GRID** enables the time reference graticule.

(4) **SPLIT** selects between analog or logic only trigger display or both (as shown here).

(5) **Analog Trigger Display** shows the analog waveform with the trigger point aligned at the center of the display.

(6) **Logic Trigger Display** shows the 8 logic signals time-aligned to the analog waveform.

(7) **Trigger Level** analog trigger level (a slider) or logic trigger condition (a button array).

(8) **Trigger Condition** displays the trigger level in volts or the logic trigger condition.

(9) **Pre-Trigger Capture** duration captured before the trigger point (as a percentage of the total).

(10) **Trigger Source** three buttons to select which of the analog channel A or B, or the logic bus is to be the trigger source. The source channel is color coded (yellow, green and blue).

(11) **Trigger Type** three buttons to select the type of trigger to apply (RISE, FALL and STATE).

(12) **Trigger Filter** three buttons to select the type of trigger filter to apply.

(13) **Hold-Off Control** when an ADJUST trigger is selected, this slider activates and allows the hold-off time to be specified.

Trigger Source (10)

Any analog channel or (combination of) logic channels can be selected as the trigger source.

The top row of channel colored buttons selects the trigger source.

Depending on the BitScope model, some **virtual instruments** do not allow all sources to be selected and it may be necessary to select a different VI to be able to select certain input channels on these devices.

For example on **BS50U** the LOGIC VI does not allow analog trigger sources to be selected.
Disallowed sources are shown "greyed out" (ie, they cannot be selected). On some older models (eg, **BS220**) the analog trigger sources are handled differently (they don't have a comparator).

Source selection on more recent models (eg, **BS310U** or **BS442N**) is unlimited.

Trigger Type (11)

There are three types of trigger condition:

- **RISE** - trigger on condition transition FALSE to TRUE.
- **FALL** - trigger on condition transition TRUE to FALSE.
- **STATE** - trigger at any time the condition is TRUE.

The second row of buttons selects trigger condition to use.

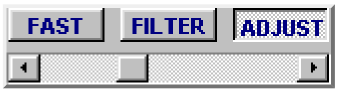
The first two are "edge triggers"; that is they trigger on the transition from one state to another.
The third is a "state trigger" it triggers at any time the trigger condition is seen to be true.

The STATE trigger is most useful when performing one-shot logic or mixed signal data capture when you have control over the source of the trigger signal. In normal (ie, repeating trace) oscilloscope applications, RISE or FALL are likely to give you better results.

Trigger Filter (12)

There are three trigger filter settings:

- **FAST** - highest speed trigger (no filter).
- **FILTER** - noise rejecting (5uS) filter.
- **ADJUST** - adjustable hold-off filter (3uS to 150uS).



BitScope's trigger circuitry is very fast, particularly when edge trigger is selected.

As a consequence you may need to choose FILTER or ADJUST to apply a filter to slow the trigger down and prevent triggers occuring on high frequency noise the may be present on analog waveforms or glitches that may appear on (or between) logic channels.

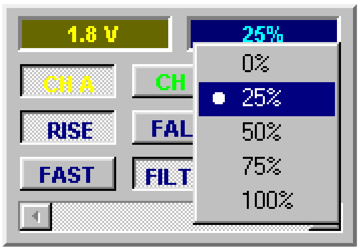
Under most circumstances selecting FILTER will be sufficient and quite reliable. For slower or lower frequency signals it may be necessary to select ADJUST and move the slider to the right until a stable trigger is achieved.

By contrast if you find you cannot get BitScope to trigger, particularly when looking at very fast logic or high frequency waveforms, be sure to select FAST to free to trigger to capture fast edges.

Pre-Trigger Capture (9)

By default DSO captures analog waveforms and logic that occurs after the trigger.

However fast clock waveform and logic capture *before* the trigger is also possible.



The position of the trigger in the captured data is selected from a standard set of 0%, 25%, 50%, 75% and 100% of the buffer:

- If set to 0% all capture is post-trigger.
- If set to 100% all capture is pre-trigger.
- If set in-between, the ratio pre/post is as specified.

The **waveform offset** operates as normal allowing the trigger point to be located anywhere on the display.

The prevailing pre-trigger setting is shown in blue directly below the trigger display. It may be changed by left-clicking (to select the next value) or right-clicking (to pop up a menu and select a value directly, as shown here).

When set to other than 0% or 100% a vertical grey marker appears on the display to show where the trigger point is in the waveform. If the graticule is enabled it may not be visible (ie, it may be hidden by the graticule). To see it, disable the graticule or scroll the waveform a little.

Pre-trigger capture will be automatically disabled if you choose parameters that are incompatible with it. For example, enabling the delayed timebase disables pre-trigger display. If the device does not support pre-trigger it will also be disabled. This applies to early models only: BS220. When the pre-trigger is disabled the pre-trigger display widget is "greyed out" it indicate this.

AUTO Trigger (2)

The AUTO Trigger makes it easy to see waveforms before you have set a trigger, when triggering is not reliable (eg, DC voltages) or when the signal changes form significantly.

AUTO forces a trigger after a timeout if no trigger condition has been seen which ensures that within a short time (twice the selected **frame period**) a trace will appear on the display.

To enable auto trigger, simply select the AUTO button.

AUTO is most useful for repeating displays with relatively high display refresh rates and reasonable fast timebase settings. If the timebase is too slow or the display refresh is low, AUTO traces can become difficult to distinguish from triggered traces (ie, "did I get a trigger event or not ?")

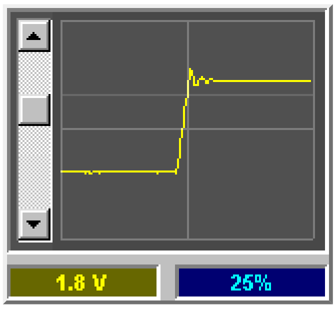
It is therefore generally a good idea to *disable* AUTO when performing slow repeating captures and especially when performing one-shot waveform or logic captures.

Triggering on Analog Signals

Analog triggers are the most familiar; simply select a source and set a trigger level.

Figure **[2]** shows a rising edge analog trigger set at 1.8V with 25% of the waveform capture pre-trigger (ie, 75% after the trigger).

The trigger level is adjusted with the slider. The cross-hairs (if GRID is enabled) show the trigger point (vertical) and zero volt level (horizontal). The second (feint, horizontal) line indicates the current trigger level and moves as the slider is adjusted.



Analog Trigger [2]

Both the waveform and the trigger level are shown in YELLOW which indicates the trigger source is Channel A (in this example).

BitScope's analog trigger circuit is applied at the A/D convertor input. This means that the trigger's sensitivity is optimized for the waveform as it is captured. This is very important, particularly for low level signals or when input prescalers are enabled. It also means that if you change the input range of the channel that is the trigger source, the trigger level may need to be readjusted.

Triggering on Logic Signals

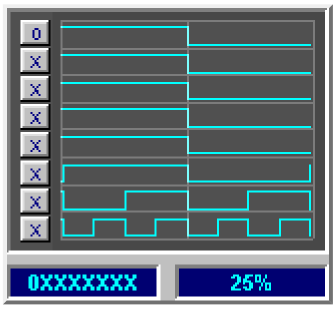
In contrast to the simplicity of analog triggers, logic triggers can be a little confusing if you've not encountered them before.

Instead of setting a level, you specify a "bit pattern" using 8 "bit buttons" that select one of:

- **1** = HIGH LEVEL is TRUE
- **0** = LOW LEVEL is TRUE
- **X** = DON'T CARE

for each logic channel. Logic channel (bit) 7 is at the top.

The *and-combination* of these bits constitutes the trigger condition.



Logic Trigger on MSB [3]

For example, figure **[3]** shows a RISING edge logic trigger applied to the most significant bit of an 8 bit incrementing counter. We do not care what values the other logic channels have in this case.

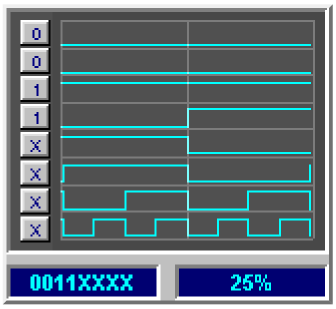
The first thing you'll notice is that no edge at the trigger point appears to be "rising".

This is a critical concept to understand concerning trigger logic: a "rising edge" trigger condition means the transition from FALSE to TRUE. In this case, at the trigger point the most significant bit (ie, the only one we care about) is switching from high (FALSE) to low (TRUE) because we set its condition to **0** (ie, LOW LEVEL is TRUE).

Similarly it is important to understand that the condition becomes true when and only when the last bit of the pattern becomes true.

Figure **[4]** shows a rising edge pattern **0011XXXX** applied to the same 8 bit counter. The condition becomes true when channel 4 goes high because at this point in time channels 7 and 6 are already low (TRUE) and channel 5 is already high (TRUE). Again, the levels on channels 3, 2, 1 and 0 are irrelvant to the trigger condition.

If you select FALLING edge, the trigger occurs when the bit pattern is seen to be TRUE and then *become* FALSE.



Logic Trigger on 4 bits [4]

If you select STATE trigger, operation is similar to a rising edge but without the "edge". In this case the trigger occurs as soon as the condition is seen to be true but unlike a rising edge trigger it does not need to be seen to be false first. That is, it does not need to be a transition from FALSE to TRUE, it simply needs to be seen to be TRUE. The difference is subtle but important in some situations, particularly when triggering on a repeating waveform or logic patterns where you need a stable repeating trigger.

In summary, STATE logic triggers are the easiest to understand (you simply define the "truth" condition) but edge logic triggers are often more useful (they see state transitions, not just states).

Using Digital Triggers on Analog Signals

An alternative to the analog trigger (as applied to an analog waveform) is the digital trigger.

The digital trigger is essentially the same as the logic trigger but applied to the digitally encoded analog waveform. It can take some getting used to but it's potentially very useful when used with trigger filters because it allows you to set up trigger conditions for certain complex waveforms that are almost impossible to define any other way.

The digital trigger is selected by choosing an analog trigger source (eg CH A) with a STATE trigger.

The trigger level slider is replaced with a set of bit buttons but unlike the logic trigger, the waveform display continues to show the analog waveform.

Digital Trigger [5]

As you set up the bitwise trigger condition, one or more grey bands appear on the waveform display indicating the levels at which the trigger is true and the darker areas where the bands do not appear show the levels where the trigger is false.

Fig **[5]** shows a digital trigger band set up to trigger on channel A at the very peak of the overshoot on a rising edge. Because the signal reaches this point only very briefly and only once per period, using a FAST Digital Trigger can produce a very reliable trace.

Depending on how you set up the trigger, there may be more than one "truth band" and the height of the band may be anywhere from full screen to a single line (ie, single level).

Whenever the signal passes into a true band, the trigger condition will be met subject to any trigger filter that may be applied. Using the adjustable filter then, you can specify a minimum period of time the signal must remain near a given level before the trigger condition is met.

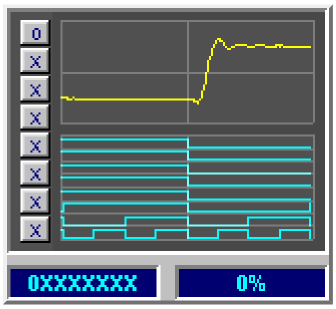
Cross-triggering on Analog or Logic Signals

On of the most powerful and useful features of BitScope is its mixed signal capture capabilities.

At its heart is the cross-trigger; the ability to trigger analog waveforms on logic and vice versa.

At it's simplest it means you can use any logic input as an external trigger source for analog waveforms. In more sophisticated setups it means you can trigger a logic capture when a related analog signal reaches a certain level or crosses a defined threshold.

In essence the cross-trigger is simply one of the previously described analog or logic triggers but used to trigger the capture of the other (or both) types of data.



Logic Cross Trigger [6]

The SPLIT button allows you to see what is happening; it splits the trigger display showing both analog and logic data and the timing relationship between them around the trigger point.

Fig **[6]** shows a logic cross-trigger of our 8 bit counter example, set to fire on the the MSB transition from HIGH to LOW (ie, the counter wrap-around)

The associated analog waveform is the counter's digital output applied to an 8 bit D/A convertor. The cross-trigger shows the timing relationship between them; the analog response is close behind the counter wrap point.

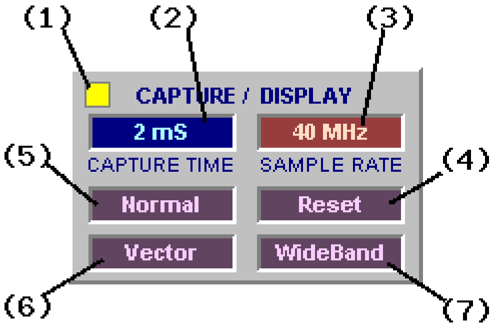
Of course it is possible to see the same detail in the main display when using the MIXED VI but the convenience of the trigger view showing this information becomes apparent when using other VIs or when examining regions in the waveform and logic other than the trigger point.

CAPTURE CONTROL

The capture control defines how waveforms and logic are acquired

BitScope usually captures at the highest sample rate it can to maximize sample resolution.

However depending on the timebase, the display sample rate is often lower which allows non-interpolating **timebase zoom**, resolution enhancement and high bandwidth displays.



The CAPTURE/DISPLAY panel establishes the parameters used to control how the BitScope captures and displays waveform and logic data.

(1) **Capture LED** illuminates in the colour of the selected trigger channel whenever the DSO is communicating with the BitScope device.

(2) **Capture Duration** shows the duration of the capture data. It is usually at least twice as long as the display duration which allows **waveform offset**.

(3) **Sample Rate** shows the sample rate used by BitScope during the most recent data capture.

(4) **Data Recorder** controls the integrated DSO data capture recorder/replayer (DDR).

(5) **Phosphor Mode** selects whether to accumulate waveforms on the display phosphor or not.

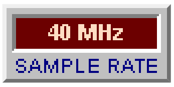
(6) **Display Mode** selects how the DSO renders waveforms and logic on the display.

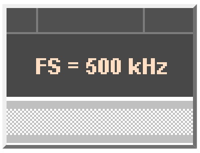
(7) **Data Mode** defines how the data is (pre)processed by BitScope (if at all).

Sample Rate (3)

There are two sample rates managed by the DSO:

1. **Capture Rate** - rate at which data is captured.
2. **Display Rate** - rate at which waveforms are displayed.





The capture sample rate is shown in the CAPTURE/DISPLAY panel (3) and the display sample rate on the display itself as "FS = ".

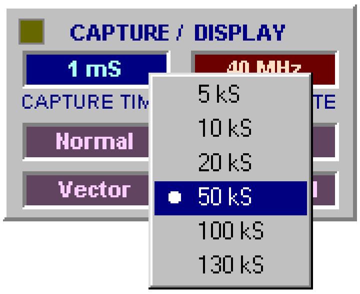
As the example to the right shows the capture sample rate (in this case, 40 MHz) is usually much higher than the display sample rate (500 kHz).

The ratio between these two rates depends on a number of factors including the selected **buffer size**, the **timebase** and timebase **zoom** and it determines how much additional resolution the Enhance **Data Mode** can realize.

Using bandlimited filtering and decimation, BitScope and DSO exploit this difference whenever possible to maximize capture and display resolution. For example, see **Quantization** below.

Capture Duration & Buffer Size (2)

The capture duration is used to select the buffer size used when capturing a data frame.



It may be changed by right-clicking the capture duration (to pop up a menu and select a buffer size directly). The range of values available depends on the connected BitScope model.

Which size you choose depends on what you want to achieve.

Smaller sizes are best suited to fast refresh analog waveform display or **persistent phosphor** (sampling oscilloscope) applications and larger values are better suited for storage oscilloscope, mixed signal and logic analyzer applications.

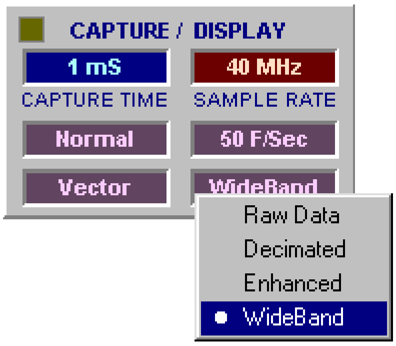
The value you choose is maintained per **instrument** so selecting a different instrument may change the buffer size.

When doing one-shot captures typical of logic analyzer work you will often want to set the capture buffer as big as it will go so you can see as much information as possible.

Another reason to use a larger buffer size is when looking at low level or high bandwidth analog signals. A large buffer displayed in the Enhanced or WideBand **data mode** affords the highest capture precision possible. In some cases you can achieve an additional 4 bits capture resolution per analog sample (ie, a total of 12 bits).

Data Mode (7)

The data mode defines how the data is (pre)processed by BitScope (if at all).



It may be changed by left-clicking the current value (to select the next value) or right-clicking it (to pop up a menu and select a value directly, as shown here).

There are four supported data modes:

1. **Raw Data** - no signal processing, just the data.
2. **Decimated** - sub-sampled data upload.
3. **Enhanced** - decimated with improved resolution.
4. **WideBand** - enhanced plus wideband data.

Which mode you choose depends on what you want to see.

Raw Data Mode

To see all the waveform or logic data as captured by BitScope, choose **Raw Data**.

Signal (pre)processing is *not* performed in this mode and depending on the **buffer size** and other factors it may be quite slow because it requires a lot of data to be sent from BitScope to the PC.

If you plan to export the data to a file for processing by other software **Raw Data** is usually the preferred mode because it saves the data as originally captured by BitScope.

It can also be useful when looking for analog glitches or very short duration logic events in one-shot data because every sample captured is shown on the display.

Decimated Data Mode

For the fastest repeating display refresh rates choose **Decimated** but beware of aliases...

BitScope usually captures at sample samples rates much higher the display's sample rate (except when using very fast timebase settings). As a consequence decimating the data in BitScope (ie, skipping samples) before sending it to the PC reduces the link bandwidth for much faster update.

As with any digital sampling oscilloscope, decimation can produce aliases; images of the waveform that appear at the wrong frequency. This may or may not be what you want and it occurs when frequencies in the signal are higher than 1/2 of the *display sample rate*.

You're unlikely to want to see aliases when looking at arbitrary analog waveforms and certainly not when viewing logic data. It's easy to tell whether what you're seeing is an alias; select **Enhanced** or **WideBand** display mode instead and if the waveform changes shape or frequency, you've been looking at an alias in Decimated mode.

Viewing aliases in Decimate mode is *useful* when sub-sampling or accumulating X-Y plots or dot displays on persistent phosphor. Using this technique it is possible to view periodic waveforms of frequencies up to the analog bandwidth limit of BitScope (100 MHz) and similar applications (eg, generating eye patterns for ISI measurements).

Enhanced Data Mode

For the highest bit-resolution capture with minimal chance of aliases choose **Enhanced** mode.

Depending on the ratio between the capture and display **sample rate** Enhanced mode can improve the bit resolution of captured data (at the display sample rate) by up to 4 bits (at total of 12 bits resolution for analog waveforms). Like Decimated mode, Enhanced mode is also quite fast.

Due to the higher bit resolution Enhanced mode can be very effective when viewing low level signals. It can also significantly enhance spectrum displays, particularly when low level periodic signals are buried in wide band noise (a common situation).

To maximize the potential benefit of Enhanced mode, choose a larger **buffer size**.

WideBand Data Mode

To see the "full picture" choose **WideBand** mode.

Like Enhanced mode, WideBand offers high bit-resolution analog waveform display but in addition it simultaneously shows high frequency components (noise or other signals) that may be present on the waveform.

Fig **[1]** shows a WideBand display of a square wave in the presence of a higher frequency tone burst and a lot of noise. Without this mode aliasing effects would distort the displayed waveform beyond recognition.

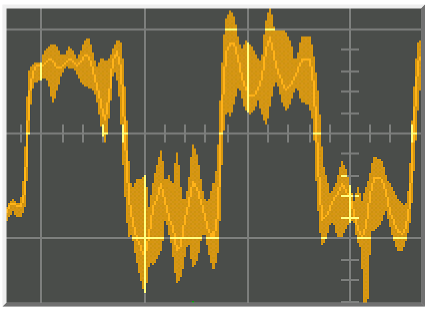


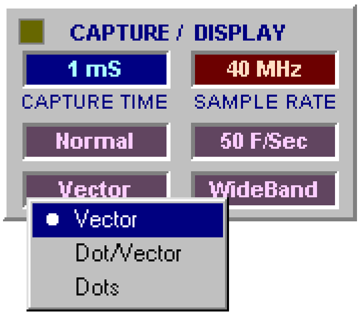
Fig [1] - High Bandwidth Waveform Display

Because this type of display is so useful it is the default display mode of most of the DSO's analog **instruments**. For anyone with experience of high performance analog oscilloscopes, WideBand should look quite familiar.

Display Mode (6)

The display mode selects how the DSO renders waveforms and logic on the display.

It may be changed by left-clicking the current value (to select the next value) or right-clicking it (to pop up a menu and select a value directly, as shown here).



There are three supported display modes:

1. **Vector** - curve fitting line draw (optionally interpolating).
2. **Dot/Vector** - sample plots with piecewise line segments.
3. **Dots** - sample plots only (sub-sampling, phosphor use).

Which mode you choose depends on which features of a waveform or logic trace you want to see and the **data mode** you've chosen.

Vector Display Mode

The default and most familiar display mode is **Vector**.

Analog waveforms are drawn as curves that fit the sampled analog data and logic traces are drawn as familiar high/low line segments.

Implicit in Vector mode is the assumption that the waveform data is sampled real-time (ie, it's not sub-sampled or equivalent time sampled). It uses bandlimited interpolation to render waveforms. This means that very high frequency real-time waveforms will be displayed as approximations.

If higher resolution rendering of high frequencies is required, equivalent time sampling (periodic sub-sampling) should be used. Vector display mode can accomodate this but Dot mode on a persistent phosphor may produce higher resolution results.

Vector spectrum displays are drawn as area filled interpolated curves.

Dot/Vector Display Mode

Similar to Vector but drawn without bandlimited interpolation and showing individual samples.

Dot/Vector is a finer display mode (it doubles the display sample rate) and allows you to see precise sample values as well as line segment transitions between them.

Dot/Vector spectrum displays are also double resolution. This can be quite useful if you need more display bandwidth to help identify aliased spectrum components or make more precise frequency or bandwidth **measurements**.

Dot Display Mode

Dot mode plots just the samples themselves. It assumes nothing about their mode of capture.

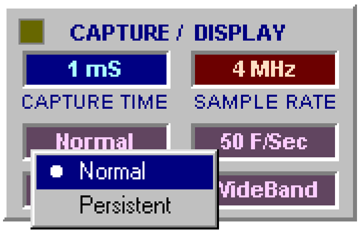
This mode is ideal for accumulating X-Y plots, sub-sampled waveform displays. It is most often used with **persistent phosphor**.

Spectrum and Logic displays are drawn the same as in Dot/Vector mode.

Phosphor Mode (5)

The phosphor mode selects whether to "accumulate" waveforms and logic on the display phosphor.

It may be changed by left-clicking the current value (to select the next value) or right-clicking it (to pop up a menu and select a value directly, as shown here).



There are three supported phosphor modes:

1. **Normal** - live display with periodic refresh.
2. **Persistent** - persistent accumulating display.
3. **Decay** - semi-persistent accumulating display.

In most situations Normal mode is used. In this mode the waveform is redrawn every frame.

However is some display and data modes, accumulating displays can be very useful to view waveform and spectral features not otherwise readily seen. Examples include Dot Display Mode X-Y plots, timing jitter displays for both analog and logic edges, eye patterns and spectral plots.

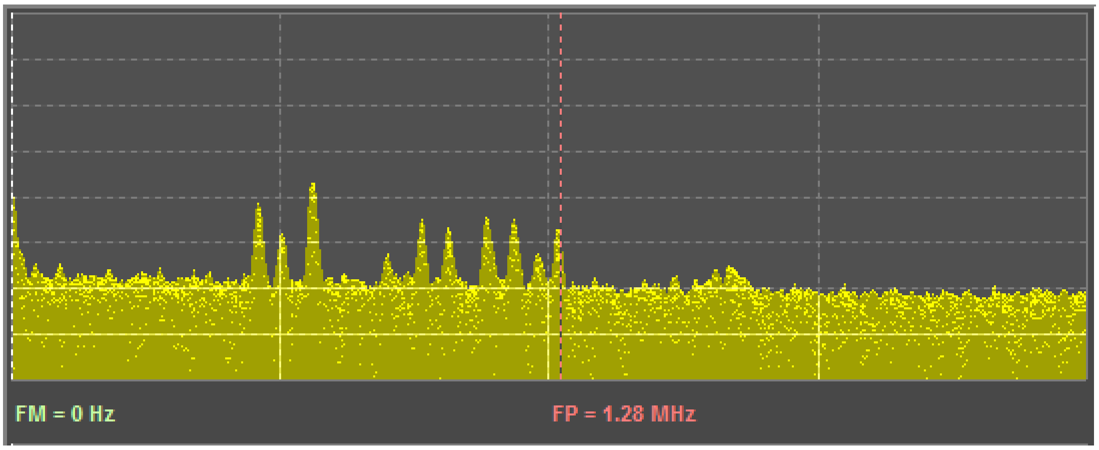


Fig [2] Sydney Radio Stations - Persistent Phosphor

Fig **[2]** shows a persistent mode spectrum display showing the peaks of a number of local radio stations in Sydney, Australia. This image was created by simply hooking a loop of wire to the input of a BS310U and enabling the input prescaler to provide sufficient gain.

Quantization Noise and Resolution Enhancement

As with any digital oscilloscope, quantization noise is produced when converting analog signals to their digital form. Normally this is not a problem because the quantization noise level is very low compared to the signal itself can cannot be seen on the display.

However when measuring very low level signals, quantization noise can become significant.

Fig **[3]** shows a 2 kHz sinusoidal waveform of very low level (about 13.7 mV peak-to-peak).

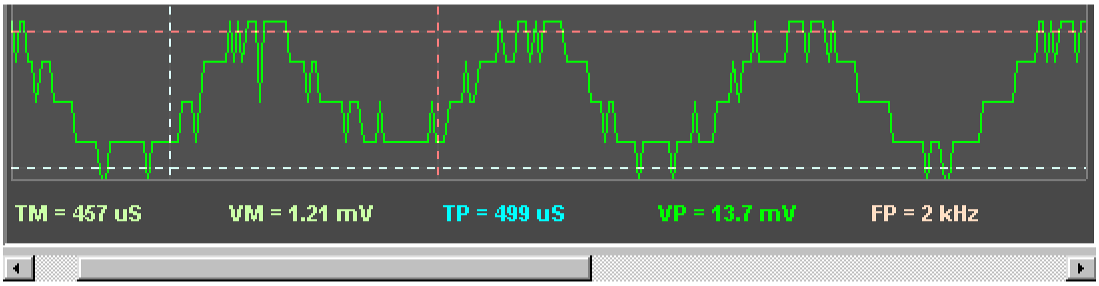


Fig [3] Small Signal - Decimated Data Mode

Even at the most sensitive **range** the level of quantization noise is significant for such a small signal. In this case it is 4mV/quantum and is clearly visible on this Decimated waveform display.

Fig **[4]** shows the same 2 kHz sinusoidal waveform but with Enhanced **Data Mode** selected.

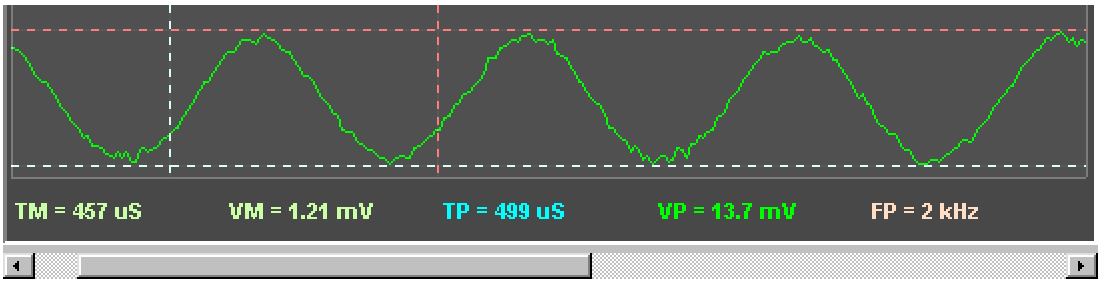


Fig [4] Small Signal - Enhanced Data Mode

The bit-resolution has been increased (by about 3 bits in this example) so the effects of quantization noise are greatly reduced and the waveform is much clearer.

Enhanced mode filters broadband noise (of which quantization noise is just one example) but you may want to know whether such noise exists in the signal even though you also want to see a high quality rendering of the waveform itself.

In this case use WideBand Data Mode. Fig **[5]** shows the same waveform in rendered WideBand.

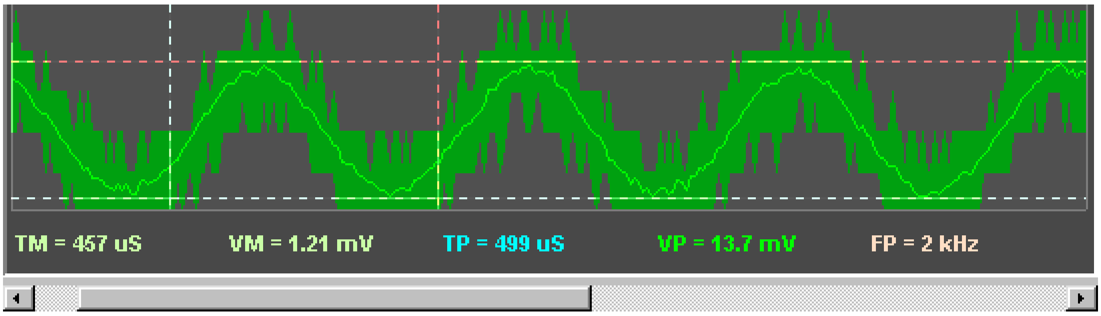


Fig [5] Small Signal - WideBand Data Mode

Here it is possible to see both the resolution enhanced sinusoidal waveform *and* the broadband (quantization) noise at the same time. In practice if the primary source of noise is quantization it's probably more useful to use Enhanced mode but where the noise comes from other sources (details of which you may need to know), WideBand makes it easy to see.

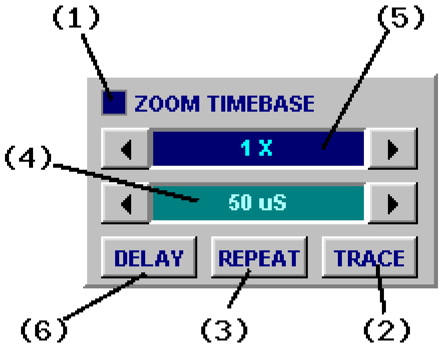
BitScope models that support **analog prescalers** can display even lower level signals and the resolution of these displays can also be enhanced using the same technique described here.

MAIN TIMEBASE

The timebase set the time scale used for capture and display

Most DSO **instruments** are *dual timebase* where the scale of each timebase is also "zoomable".

The MAIN TIMEBASE panel is where you set the main timebase and adjust the timebase zoom and is described here. The DELAY TIMEBASE (ie second timebase) appears on the **delay panel** but its zoom factor is controlled from the main timebase panel (when selected) as well.



The main timebase panel sets the "seconds per division" used to display waveform and logic on the main display.

It is also where the TRACE/REPEAT buttons that initiate capture are located.

(1) **Timebase LED** illuminates when the main timebase is selected.

(2) **TRACE Button** initiates data capture.

(3) **REPEAT Button** selects repeating capture. Also sets the (maximum) display frame rate.

(4) **Timebase Control** sets the timebase value in "seconds per division".

(5) **Zoom Control** sets timebase scaling factor (aka "timebase zoom factor").

(6) **Delay Button** enables the delay timebase for alternate/delay timebase control.

Trace and Repeat (2) (3)

The TRACE and REPEAT buttons are at the heart of all DSO **instrument** operations.

Clicking the TRACE button initiates a single ("one-shot") mixed signal data capture or **DDR replay**.

Once captured the waveform and logic data may be inspected in detail using the **timebase zoom** and **waveform offset**. The **vertical scaling** can be adjusted and **cursor measurements** made.

Clicking the REPEAT button does the same thing, only this time capture or replay and display is repeated automatically at the selected **frame rate**.

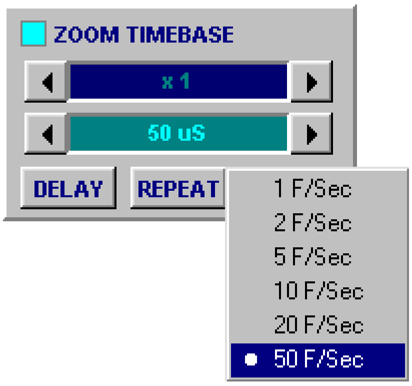
The **auto trigger** can also be useful for repeating captures.

Frame Rate (3)

Sets fastest possible refresh rate used for repeating capture.

It also sets the **AUTO Trigger** timeout.

It may be changed by right-clicking the REPEAT button to pop up a menu and select the frame rate directly.



Whether the DSO actually updates the display at this rate depends on a range of factors:

1. USB, Ethernet or RS-232 link speed.
2. Selected **timebase** and **buffer size**.
3. Selected **instrument** and enabled channels.
4. Speed of your PC and graphics card.
5. Selected **data mode** and **display mode**.

When using a PC with a 400 MHz Pentium II or earlier CPU and/or an older graphics card the maximum display refresh be be limited to below the selectable maximum (50Hz).

If the speed of the link between the PC and BitScope is slow is may also be necessary to slow the frame rate (to avoid premature AUTO trigger timeouts). An example of when this may be necessary is talking to a **Network BitScope** half way around the planet **via the Internet**.

Timebase Control (4)

The main timebase sets the "time per division", like most oscilloscopes.

The timebase value used for waveforms as displayed is always shown on the display (Fig **[2]**).

The chosen timebase implicitly sets the sample rate used for data capture; the DSO always trys to use the highest sample rate it can based on the current **buffer size** to capture waveforms at the selected timebase. This maximizes the non-interpolating **Timebase Zoom** available.

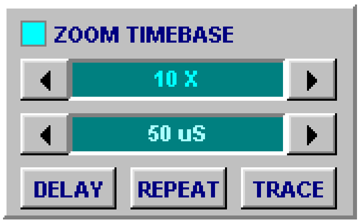
Timebase Zoom (1) (5)

All BitScopes can capture far more data than can be shown on the display at once.

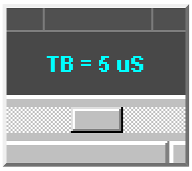
Timebase Zoom takes advantage of this and allows you to "zoom in" on fine waveform details or "zoom out" to see an overview of the entire waveform on screen all at once.

By default most DSO **instruments** scale the display to show 1/2 of the total capture buffer. When set other than x1 the timebase zoom parameter (5) lights up and this ratio changes.

Fig **[1]** shows the main timebase zoomed by a factor of x10 resulting in a zoomed timebase of 5 uS/Div.



Zoom Timebase [1]



Timebase Value [2]

The zoomed timebase value is shown at the bottom of the main display for a given capture, in this example shown in Fig **[2]**.

In fact whatever combination of timebase (main or delay) and timebase zoom you choose, the resulting timebase value for a given capture will always been shown at the bottom of the display.

The other point to note is that the **waveform offset** slider scales in size according to the timebase zoom selected.

In this case its width it is 1/2 x 1/10 = 1/20th of the display width indicating that for timebase zoom of x10 you can scroll the waveform left and right for a total of 20x the display width.

Grid Off [3]

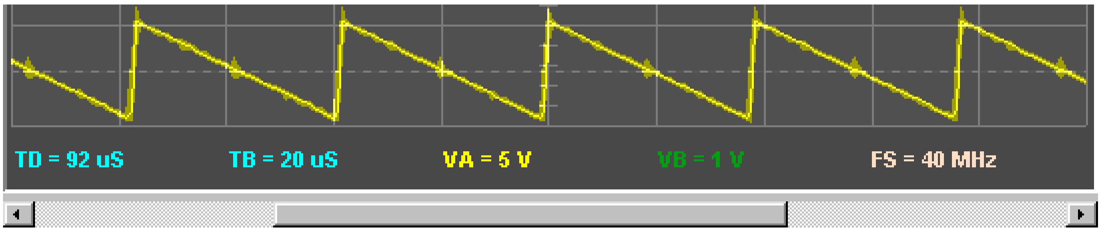
Note also that if you disable the main display grid the timebase value switches to report "time per display" instead of "time per division". Fig **[3]** shows this as 50uS because the time per division was 5uS and there are 10 divisions per display.

Waveform Offset

DSO can scroll through the capture buffer to move waveform to where you want it on display.

Together with **timebase zoom**, waveform offset provide very powerful means by which you can choose which part of a captured waveform or logic trace you want to see and in how much detail.

Scrolling the waveform means to apply a time **offset** to the waveform:



Waveform Offset Control [4]

The scroll bar below the main display is used to adjust the offset that is to be applied (Fig **[4]**). In this case a 92 uS offset has been applied to locate a rising edge at the middle of the display.

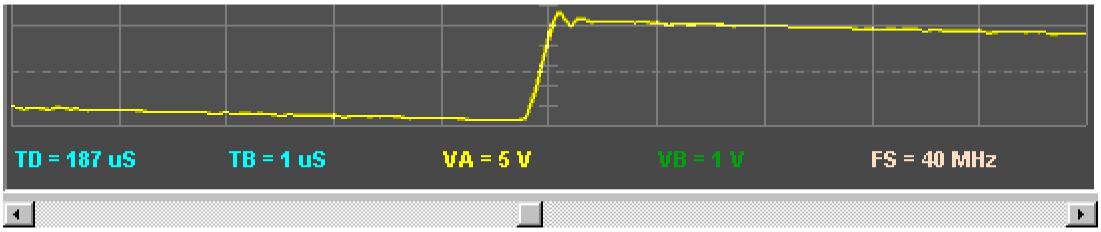
The offset is reported at the lower left edge of the display (shown as TD = 92 uS in this example).

If the **delay timebase** is enabled, the value reported is the sum of the post-trigger delay and the offset. If a non-zero **pre-trigger** is selected, the value reported may be negative.

If you need to know precisely how much of the reported value is due to the waveform offset (ie, scroll bar position) you can see this reported as the second value in the **delay timebase panel**.

Zoomed Waveform Offset

When a feature has been located to the center of the display it can be "zoomed".



Zoomed Waveform Offset [5]

Fig **[5]** shows the same waveform as Fig [4] with no change made to the waveform position but the **timebase zoom** set to x20. The waveform offset is now reported 187 uS which maintains the relative position of the edge at the middle of the display.

The zoom can be increased as far as 50X which is in fact a time scaling of 100 relative to the entire buffer. If the capture sample rate is high enough and the selected buffer size large enough the waveform timescale is reduced without any data loss or interpolation and without requiring recapture (ie, it operates on one-shot captured waveforms).

DELAY TIMEBASE

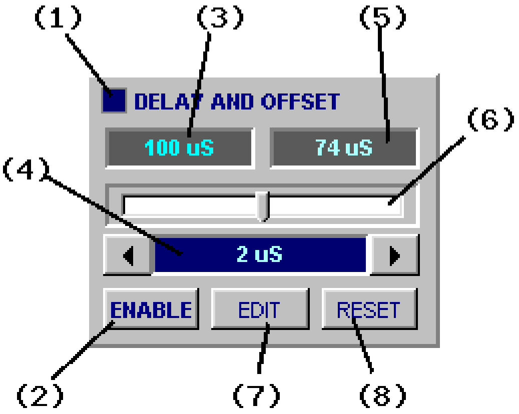
The delay timebase controls delayed waveform capture

Most DSO **instruments** have a second *delay timebase* in addition to the **main timebase**.

The delay timebase is similar to the main timebase except it starts after a pre-programmed delay.

Using the same trigger for both timebases the delay allows you to see a repeating signal "in overview" (main timebase) and also another part of it in more detail (delay timebase).

While similar to **timebase zoom** the important difference is that you can capture at a different sample rates and durations and the delay period can be much longer than a single capture buffer duration (> 2 hours if you can wait that long :-)



For example, view an entire video field in the main timebase and zoom in on the 112'th video line using the delay timebase at full resolution.

(1) **Delay LED** illuminates when delay enabled.

(2) **Delay Enable** click to enable the delay.

(3) **Delay Value** prevailing post-trigger delay.

The minimum delay depends on the BitScope model and is typically 8 uS.

For shorter delays, use **timebase zoom** together with **waveform offset**.

(4) **Delay Timebase** sets the delay timebase value. Set in "seconds per division".

(5) **Offset Value** shows the prevailing offset applied to the captured waveform.

Delay plus offset is the total time between the trigger and the left edge of the display (when delay timebase is active). The offset is adjusted via the **Waveform Offset** under the main display.

(6) **Delay Shuttle** adjusts the delay value.

This is a "shuttle control" (ie, it changes delay at a speed proportional to position).

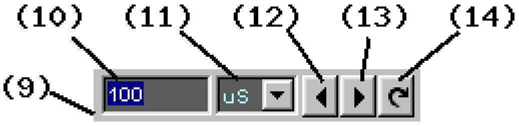
This is convenient when viewing a repeating delayed waveform in the delay timebase as you can scroll the waveform "past your eyes" until a feature of interest comes into view.

In the main timebase it adjusts the **delay band** position on the display.

(7) **Delay Editor** opens the delay editor.

(8) **Delay Reset** reset delay to the minimum.

(9) **Delay Editor** when enabled (7) the editor replaces the shuttle (6) allowing precise delay values to be entered.



Post-Trigger Delay Editor

On slower PCs or graphics cards or when a slower **frame rate** is selected the shuttle control may become more cumbersome to use.

In this case the delay editor may be more convenient.

(10) **Delay Value** editable delay value. Enter a new delay value here.

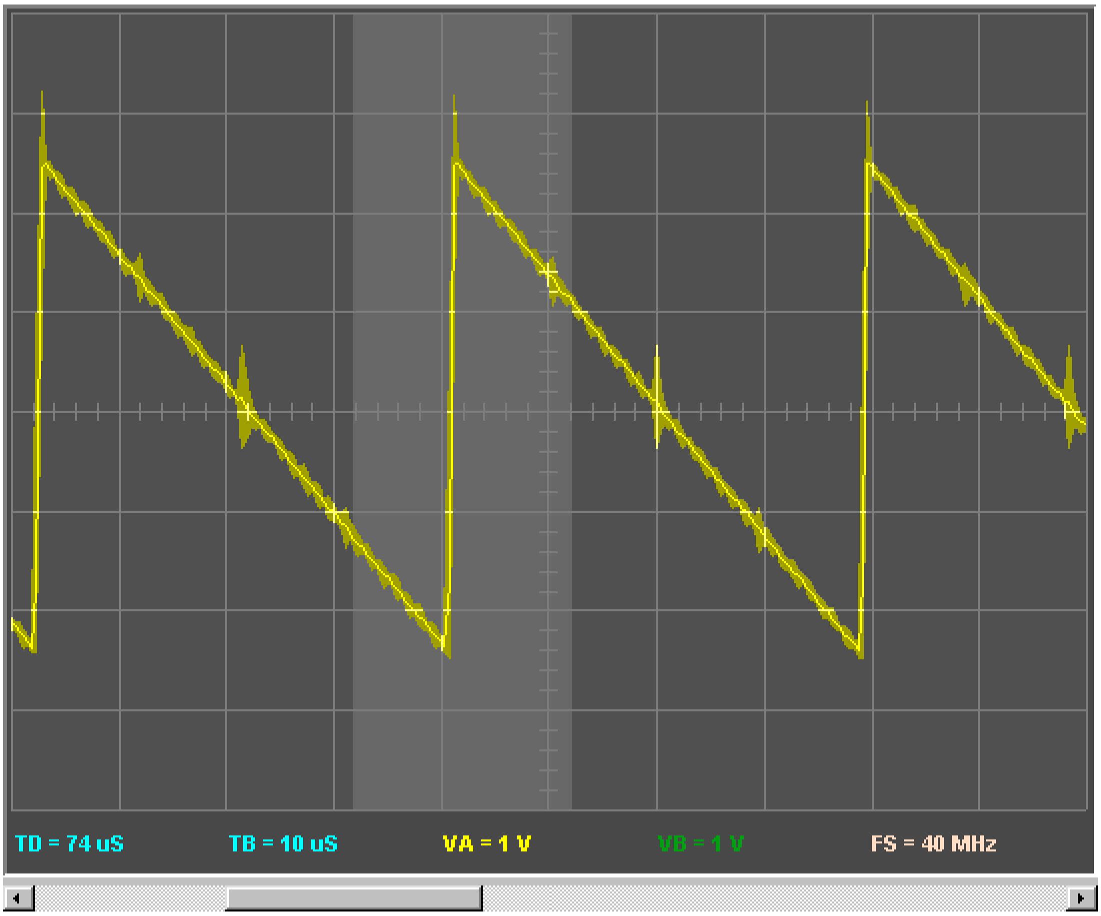
(11) **Delay Range** delay value units (uS, mS or S). Note that the BitScope itself may not support precisely the delay value you enter in the smallest (uS) range but it is always possible to adjust the waveform position down to the nearest sample using the **offset control**.

(12) **(13) Decrement/Increment** substract or add the value entered (10) from/to the current delay value (3).

(14) **Update Delay** replace the current delay (3) with the new value (10).

Delay Timebase Example (Delay Band)

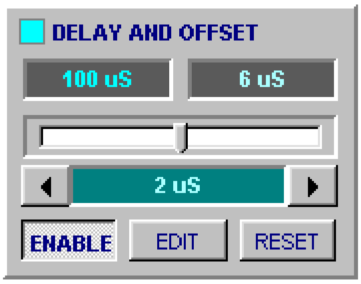
When the main timebase is active, the delay panel is enabled, and the delay within the range of the display, a delay band is shown indicating the location of delay timebase (when it's enabled).



Delay Band on Main Timebase Display [1]

Fig **[1]** shows the delay band corresponding to a delay (3) of 100 uS as shown on the 10 uS/Div main timebase display.

At first glance you may expect the delay band to be located to off the right edge of the display (because 10 divisions x 10 uS/Div = 100 uS) but the display also has a 74 uS **waveform offset** (5) applied and the delay timebase has a 6 uS waveform offset applied.



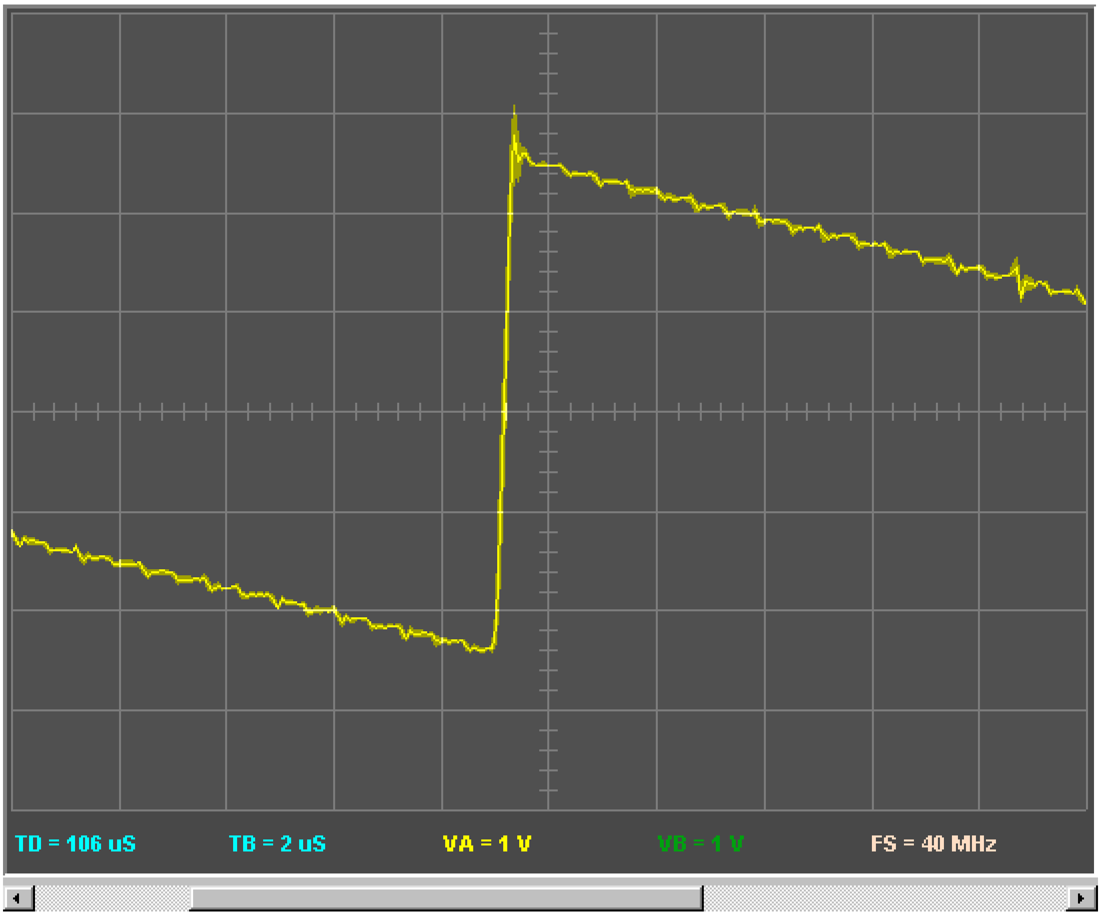
Enabled Delay Timebase [2]

Fig **[2]** shows the delay panel when the delay timebase is enabled which shows the 6 uS delay timebase waveform offset.

Note also that the ENABLE button (2) is down and the Delay Light (1) and the delay timebase value (4) both illuminated indicating the delay timebase is active instead of the main timebase.

Taken together it means the delay band starts 106 uS after the trigger and with a delay timebase value of 2 uS/Div (4) the delay band is 20 uS wide.

When the delay timebase is enabled and the waveform recaptured, the region shown in the delay band is instead displayed full screen.



Delay Timebase Display [3]

Fig **[3]** shows the same waveform as Fig [1] but this time viewed with the faster delay timebase.

Note that the display reports left edge (TD) as 106 uS (the start of the delay band shown in the main timebase above) and the delay timebase value of 2 uS/Div.

This example is relatively simple. It would have been possible to achieve similar results using the main **timebase zoom**. However when the delay required is (say) 100mS and the delay timebase set to 1uS/Div, main timebase zoom is not suitable.

A typically application for delay timebase is in mixed/logic analysis applications where some brief event you wish to capture occurs a deterministic but long period time after a trigger event.

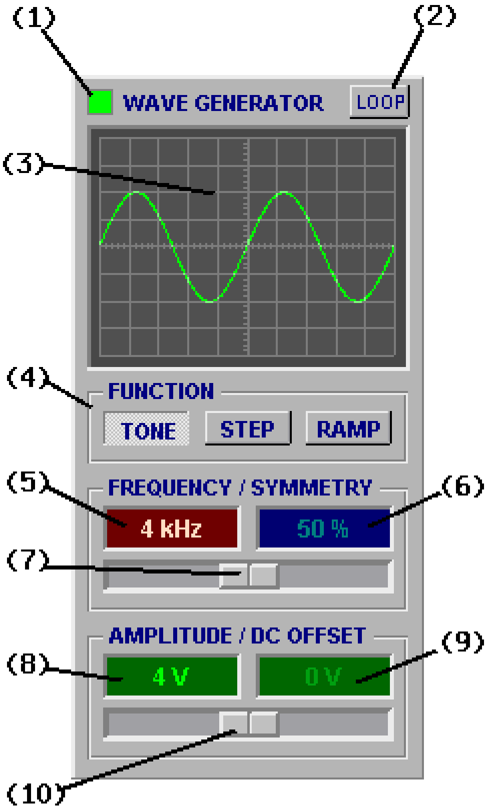
WAVEFORM GENERATOR

The waveform generator synthesizes and generates waveforms

Some BitScope models have a built-in waveform generator selected via the **AWG** instrument.

The Waveform Generator panel controls the operation of the waveform generator with parameters that will be familiar to those who have used a standard waveform generator before.

It allows the selection of a function type and wave shape (symmetry), frequency, amplitude, and offset at which it is to be synthesized.



(1) **Capture LED** illuminates when the waveform generator is active.

(2) **Generator Mode** selects whether to produce continuous waveform or one-shot waveforms.

(3) **Waveform Preview** displays generated waveform in preview.

(4) **Wave Function** selects the function used to synthesize the waveform (ie, wave shape).

Parameters are divided into two pairs each of which shares a **Parameter Control** (7) & (10):

(5) **Frequency** waveform frequency.

(6) **Symmetry** applies optional symmetry modifications to the wave shape.

(8) **Amplitude** adjusts amplitude.

(9) **Offset** adjusts DC offset.

How the waveform generator works in detail depends on the BitScope model used but they all work the same way in princple; they upload a waveform to the logic capture buffer which then replays it.

Because the logic buffer is used by the waveform genertor, concurrent logic capture is not possible.

The AWG is effectively a very high speed "sample replay engine" so the actual waveform can be completely arbitrary (within the sample rate and hardware limits of the device) and in most models simultaneous waveform capture also is possible allowing **transfer function analysis**, one of BitScope's most powerful features.

Some BitScope models also have a front panel AWG switch ->

In addition to selecting **AWG** these models require this switch to be in the ON position (as shown) for the waveform generator output to be available at the Channel B BNC connector.

This can be very useful as it switches Channel B from being an AWG output to a DSO input so with the switch off you can see whether the circuit to which you are connecting the waveform generator has any other signals, offsets or noise already present and if all is well switch it on to apply the waveform to the circuit.

Note that the AWG output is terminated in 50 ohms and an alternative waveform generator output is available on pin 26 of the POD connector (some models) regardless of the switch position.

Generator Mode (2)

The waveform generator operates in two modes:

1. **One-Shot Replay** - with optional repeat (non-loop).
2. **Continuous Replay** - continuous waveform (loop mode).

One-shot replay with optional **REPEAT** is the most useful mode when BitScope is used stand-alone.

It plays the waveform once (or many times) as a (repeating) one-shot waveform burst.

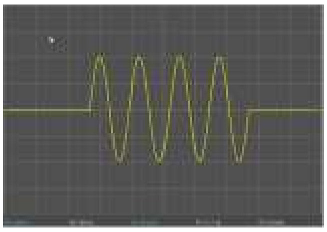


Fig [1] - One-Shot TONE

For example, **Fig [1]** shows a four cycle one-shot **TONE** function.

In one-shot mode simultaneous waveform capture via Channel A is possible and the generated waveform may be viewed on Channel B as it is replayed (ie, as an external oscilloscope would show it).

Generation and capture is synchronised to the waveform burst and any waveform generator parameter may be changed at any time.

The alternative "Loop Mode" is selected via the Loop Button at the top of the panel.

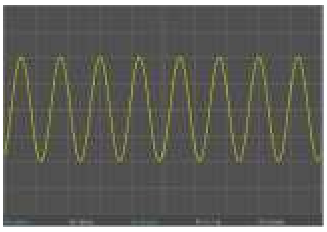


Fig [2] - Continuous TONE

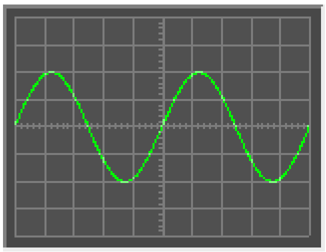
For example, **Fig [2]** shows a continuous **TONE**.

Loop Mode is normally used to produce a continous waveform where BitScope functions as a waveform generator to be used with another oscilloscope or test instrument.

Selecting loop mode precludes simultaneous waveform capture and AWG parameters may not be changed until loop mode is deselected.

Waveform Preview (3)

The waveform generator includes a mini-scope display for waveform preview.



The preview is the same as the main display except it shows the waveform that *will be* generated not the actual output waveform.

It adjusts according to the timebase and voltage scale so synthesized waveforms appear the same as they do on the main display.

The purpose of the preview is to provide instantaneous visual feedback about what the waveform will look like as waveform generator parameters are changed.

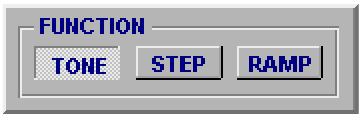
It does this even when the waveform generator is not enabled.

Of course the generated waveform can be recaptured and shown on the main display via its associated analog channel in **One-Shot Replay** generator mode.

However in **Continuous Replay** mode (aka **Loop Mode**) this is not possible so the preview display provides a useful alternative view of the generated waveform in this case.

Wave Function (4)

The first step when programming the waveform generator is to choose a wave function.



1. **TONE** - sinusoidal waveforms (pure tones).
2. **STEP** - square, pulse and impulse waveforms.
3. **RAMP** - triangle, ramp and inverse ramp waveforms.

The wave function is not the waveform. Rather it is the mathematical function used to synthesize the waveform according the specified wavform generator parameters.

TONE Function

Synthesizes a standard sinusoidal waveform (assuming symmetry of 50%).

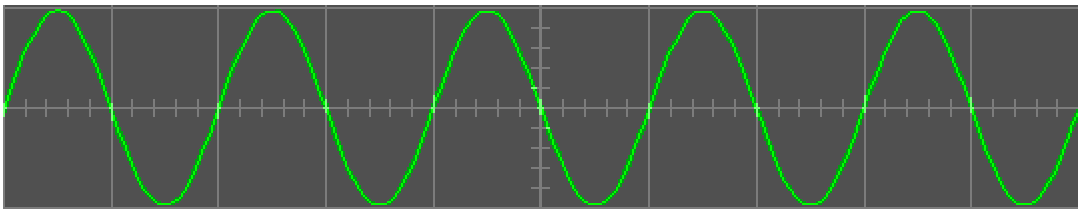


Fig [3] - TONE Wave Function (50%)

TONE function waveforms are used for tone based circuit analyses including phase response and any other application that requires a single pure frequency.

If the symmetry is set to other than 50% the sinusoid is progressively phase distorted.

STEP Function

Synthesizes a standard square wave (assuming symmetry of 50%).

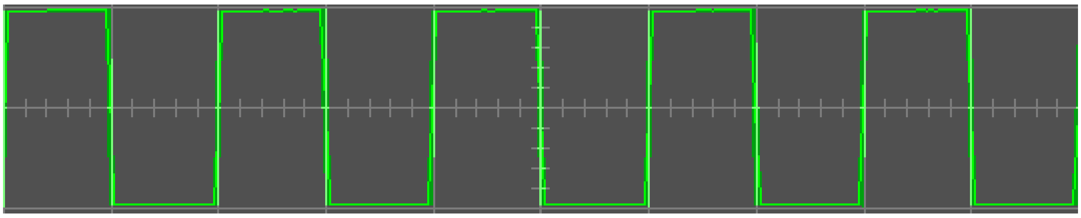


Fig [4] - STEP Wave Function (50%)

STEP function waveforms are used for step and impulse response analysis and general **transfer function analysis**. If the symmetry is adjusted to other than 50% the mark-space ratio of the square wave can be adjusted. In the extreme you can synthesize a pulse train in this way.

RAMP Function

Synthesizes a standard triangle waveform (assuming symmetry of 50%).

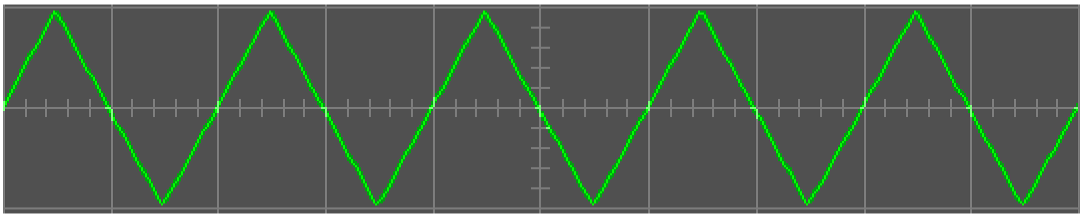
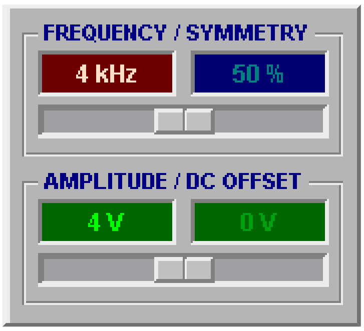


Fig [5] - RAMP Wave Function (50%)

RAMP function waveforms are useful to test system linearity, slew rate measurements or for their unique spectral characteristics. If the symmetry is adjusted to other than 50% the triangle waveform can be transformed into a ramp or inverse ramp (aka sawtooth) waveform.

Parameter Control (7) (9)

In addition to the **wave function** the waveform generator divides the four other parameters (**frequency, symmetry**, **amplitude and offset**) into two parameter groups.



Each group displays the parameter pair and has a shared parameter slider positioned below them.

To change a parameter, click on its value to highlight the parameter and adjust the slider to change the parameter.

As the parameter is changed the waveform preview adjust immediately to reflect the change to the waveform.

Keyed Parameter Entry

It is also possible type in a new value for each parameter.

Click on the parameter to be changed and type in a new value pressing *enter* to finish.

The value typed may be followed by an optional range character (eg, **m, k, M**).

For example the value **23.4k** entered into the frequency parameter box will change the frequency to *23.4kHz*. If no range is entered, units are assumed.

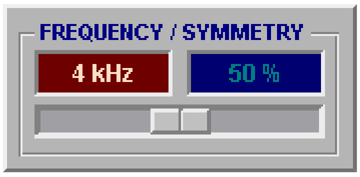
It is also possible to enter a new value with a postfix operator to apply to the parameter value.

For example, if the frequency is *4kHz* and **3k+** is entered, the new frequency will be *7kHz*.
Likewise entering **2/** will produced *2kHz* and entering **10*** will produce *40kHz*.

As new parameter values are entered, the **waveform preview** updates immediately to show what the new waveform will look like. If the previewed waveform appears too small or the timebase is inappropriate, simply adjust the associated DSO contols (**timebase** and/or **voltage scale**).

Frequency and Symmetry (5) (6)

The frequency and symmetry are the primary waveform generation parameters.



The frequency is arguably the most important because it sets the fundamental frequency of the waveform. However, when choosing the frequency it is important to understand how it relates to the prevailing capture **sample rate**.

If the sample rate is too low, the waveform will not be rendered very accurately, particularly when using the TONE or RAMP functions. The STEP function is much less sensitive to this issue but in any case the simplest way to make sure you're using the optimal sample rate is to choose a **timebase** that results in just a few periods of the waveform being displayed via channel B on the main display in one-shot mode.

See **Parameter Control** for details about how to view the waveform live.

Waveform Symmetry

The symmetry control adjusts the mark-space of STEP functions, the rise/fall ramp rate of STEP functions and the relative time span of each half of TONE functions.

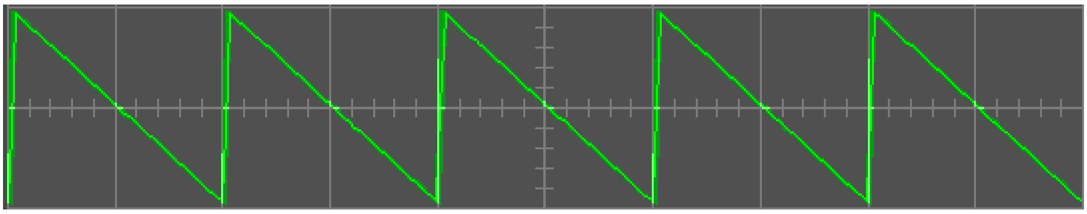


Fig [6] - RAMP Wave Function (1%)

Fig [6] shows the RAMP function with 1% symmetry; the triangle waveform has been transformed into a sawtooth. Similarly a 1% STEP function (a pulse train) is shown **Fig [7]**.

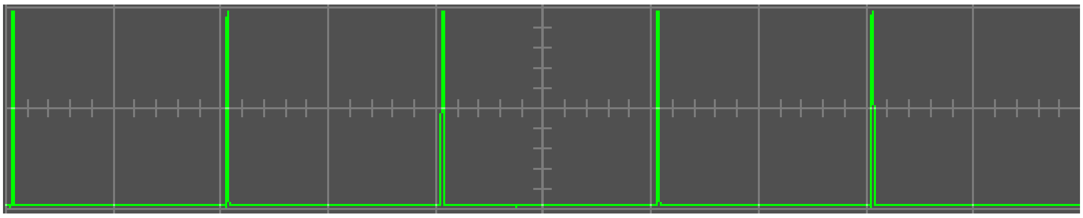
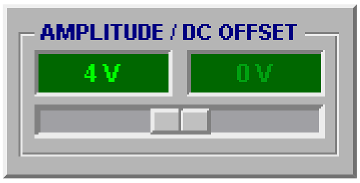


Fig [7] - STEP Wave Function (1%)

Any variation between these extremes is possible.

Amplitude and Offset (9) (10)

The amplitude and offset are the secondary waveform generator parameters.



Neither of these controls affects the waveform shape or how it is encoded for replay but they are both very important.

The amplitude controls the waveform generator's peak-to-peak output voltage swing. The largest amplitude possible depends on the BitScope model and to some extent the prevailing offset.

The offset controls the DC offset (if any) applied to the waveform. By default generated waveforms are bipolar; that is they oscillate between negative and positive voltages.

In many situations a unipolar waveform is required, for example a 0 to 5V square wave in which case the offset and amplitude controls together can be used to adjust the output accordingly.

Transfer Function Analysis

Simultaneous waveform generation and capture facilitates transfer function analysis.

By using waveforms of appropriate shape, frequency and/or spectral characteristics, BitScope can be used to characterize electronic circuits without the need for additional test equipment.

For example you can test impulse, step, amplitude and phase responses directly and using the spectrum analyzer view approximations of the associated transfer functions simultaneously.

Tank Circuit Example

Back in Electronics 101 you no doubt learnt about the LC or **tank circuit ^**.

We'll use this circuit to explain how to measure step response and view the transfer function.

Fig [8] shows a tank circuit configured to be driven by the built-in AWG (via Channel B) and the response captured via BitScope's Channel A.

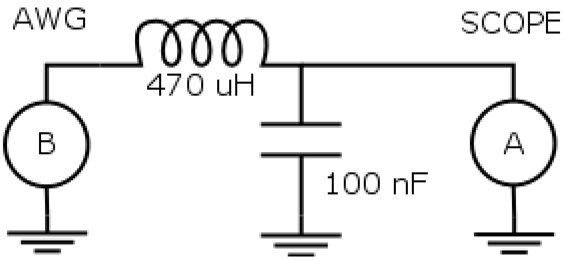


Fig [8] - Tank Circuit

If you have a spare inductor and a capactor you can build one easily. A "back of an envelope" construction of this circuit is shown in **Fig [9]**.

Fig [9] - "Back of an Envelope" Tank Circuit

It comprises a **470 uH** inductor and a **100 nF** capacitor configured per the circuit in Fig [8].

When an impulse or step is applied to this circuit it will oscillate at its "resonant frequency".

However because there is actually also some resistance (not shown in Fig [8]) this is in fact an **RLC circuit ^** so the oscillation will decay.

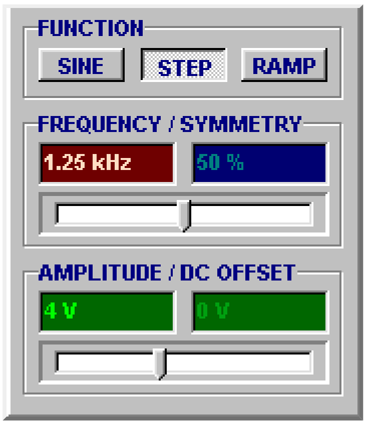
Therefore if we set up BitScope's AWG to produce a square wave and plot the result we expect to see a damped sinusoidal waveform.

The frequency of the ringing can be calculated using the standard formula, **Fig [10]**.

In this case we expect a resonant frequency of **23.2 kHz** so we'll set up the waveform generator to produce a square wave about one order of magnitude slower so we can observe a reasonable number of ringing oscillations.

$$f_{\text{resonant}} = \frac{1}{2\pi\sqrt{LC}}$$

Fig [10] - Resonant Frequency



FUNCTION
SINE STEP RAMP

FREQUENCY / SYMMETRY
1.25 kHz 50 %

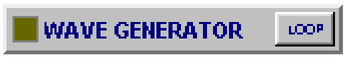
AMPLITUDE / DC OFFSET
4 V 0 V

Fig [11] - AWG Setup

Fig [11] shows the AWG set up to produce a 4V peak-to-peak square wave at 1.25 kHz.

We've left the **Symmetry** control at 50% in this case because we're looking at the step response.

If we were interested in the impulse response we'd simply set Symmetry to a small value to synthesize an impulse instead.



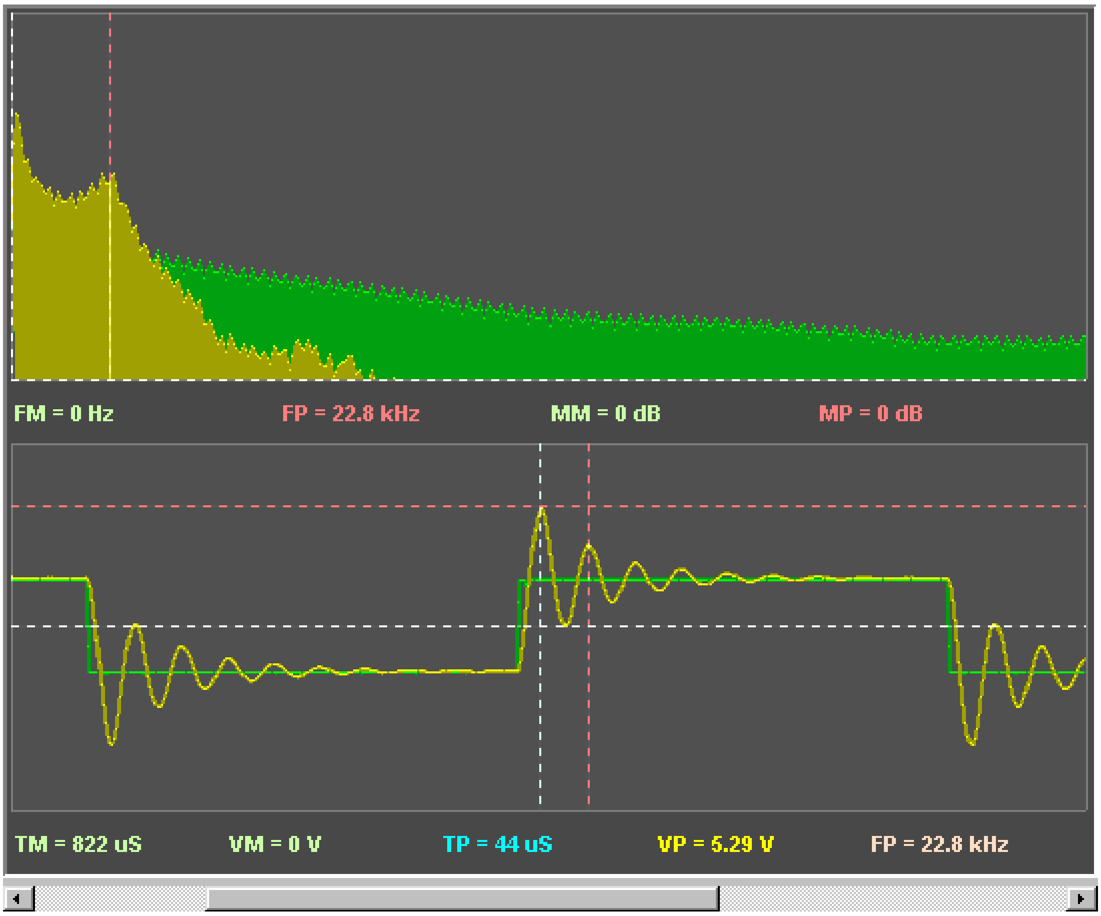
Make sure the AWG LOOP button (on the DSO software) is UP.

On the front panel of the BitScope ensure the Waveform Generator switch is DOWN and the 50 ohm terminator is UP.

These settings are shown in **Fig [12]**.

When you enable AWG, both analog channels and the spectrum analyzer you should see a display similar to **Fig [13]**.

Fig [12] - LOOP and AWG



FM = 0 Hz FP = 22.8 kHz MM = 0 dB MP = 0 dB

TM = 822 uS VM = 0 V TP = 44 uS VP = 5.29 V FP = 22.8 kHz

Fig [13] - Tank Circuit Step Response and Transfer Function

In this screenshot we've disabled the graticule and enabled the **measurement cursors**.

The green waveform and spectrum display Channel B; the Waveform Generator output. The yellow waveform and spectrum show Channel A; the Tank Circuit response.

Some features of note are:

- The AWG drives the Tank Circuit with a 1.25 kHz square switching from 2V to +2V.
- The Tank circuit response shows the classic damped sinusiodal shape we expect.
- The **time cursors** measure a ringing period of 44 uS; a frequency of 22.8 kHz.
- The ringing peak is **measured** at 5.29V, a higher voltage than the excitation waveform.
- The **frequency cursor** measures a peak at 22.8 kHz which agrees with the time domain.
- Above the resonant frequency the spectrum rolls off due to the effect of the capacitor.

The measured results match the theory to within the tolerance of the components used.

This is just one example of how to use BitScope's built-in AWG for transfer function analysis.

SPECTRUM ANALYZER

The spectrum analyzer shows baseband spectra of analog waveforms

A baseband real-time spectrum analyzer is included as part of the core DSO engine.

Its output is available in most **instruments** that deal with analog waveform data.

The baseband analyzer is integrated with the instruments it serves, so there are no additional settings to adjust.

Instead the analyzer adapts automatically to the prevailing timebase, scale, volts/div and screen size to display the optimum resolution magnitude (or phase) spectrum of the waveform.

Fig [1] shows the spectra of the signals connected to the **Sydney** BitScope as well as the waveforms themselves.

Unlike most swept spectrum analyzers the BitScope spectrum analyzers are fourier transform based and update live at the same high speed refresh rate as the waveform display.

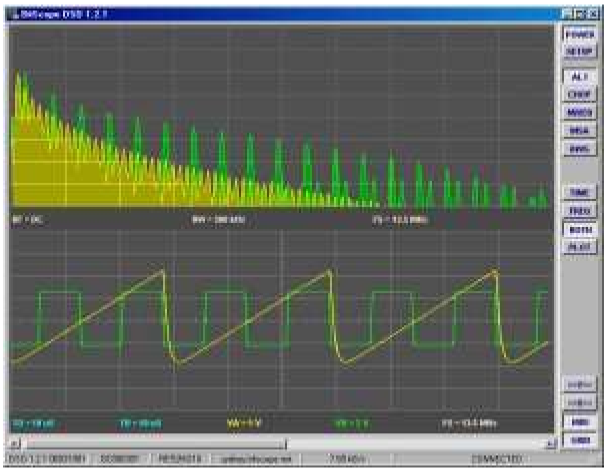


Fig [1] Baseband Spectrum Analyzer (click to enlarge)

This means it is possible to observe fast moving dynamic features in the spectrum or catch transient spectral events not otherwise visible with a swept spectrum analyzer.

It also means it is possible to zoom and scroll through one-shot waveform captures to view and analyzer the spectrum dynamically at different points in the buffer.

Spectrum parameters

Located at the bottom of the spectrum display is a set of spectrum measurement parameters:

Id	Name	Description
BF	Frequency	The frequency at the left edge of the spectrum display. In baseband this is always "DC". In multiband it's the lowest displayed frequency.
BW	Bandwidth	Spectrum bandwidth (aka "Nyquist frequency"). In baseband it is also the highest displayed frequency.
FS	Sample Rate	This reports the capture or display sample rate depending on the prevailing **data mode**; *WideBand* and *Raw Data* report the capture rate, *Enhanced* and *Decimated* report decimated display rate.
FM	Frequency Mark	The value of the mark frequency **cursor**.
FP	Frequency Point	The difference between the point and mark cursor frequencies.
MM	Magnitude Mark	The value of the mark level cursor reported in Decibel Relative units.
MP	Magnitude Point	The difference between the point and mark levels cursors.

Magnitude Scaling

The normal vertical (ie magnitude) resolution of the spectrum analyzer is 10 dB per grid line and precise dB levels can be measured using the spectrum level cursors.

The spectrum magnitude display is unreferenced; all measurements are relative to the prevailing V/Div value on each channel. If you wish to shift the spectrum up to see low level signals simply increase the vertical scale of the associated channel.

However if you wish to make measurements relative to a standard Decibel scale you can reference the display to show **dBu**, **dBV** or **dBm** values as follows:

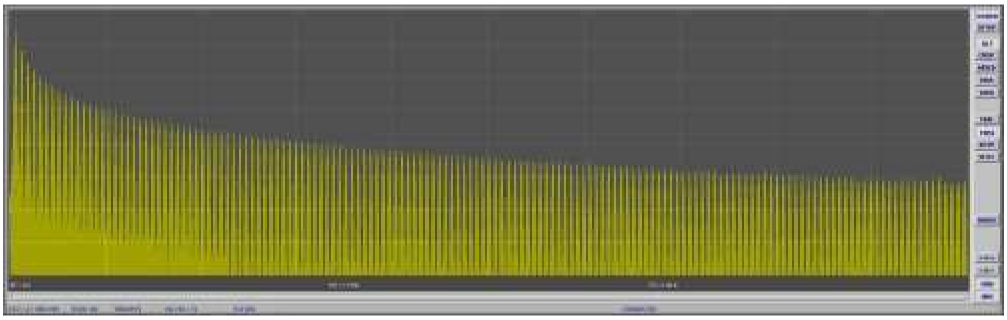
Scale	Reference	Calibration
dBu	0.775V (RMS)	Apply a 1 kHz 0.775V (RMS) voltage to any channel via a 1M Ohm (or

Scale	Reference	Calibration
		greater) probe. Mark the peak of the spectral line for **0 dBu**.
dBV	1.0V (RMS)	Apply a 1 kHz 1.0V (RMS) voltage to any channel via a 1M Ohm (or greater) probe. Mark the peak of the spectral line for **0 dBV**.
dBm	0.224V (RMS)	Apply a 1 kHz 0.224V (RMS) voltage to channel B with 50 Ohm termination enabled. Mark the peak of the spectral line for **0 dBm**.

Frequency Resolution

The frequency resolution of the spectrum analyzer depends on the bandwidth (ie the prevailing **timebase** settings) and display sample rate (which depends on display size and **display mode**).

Choose *Dot/Vector* or *Dot* modes to maximize the frequency resolution for a given display size.



If you have a "wide screen" PC, maximize the width of the DSO application to take full advantage of the higher horizontal resolution to further improve the frequency resolution. For example, the display above has been widened to show the first 153 harmonics of a 6.51kHz sawtooth waveform.

The bandwidth is reported via the **BW** parameter and the graticule has 10 vertical divisions allowing quick frequency measurement to be made in much the same way as time measurements may be made with waveform displays.

More precise measurements of spectral features can be made using **frequency cursors**.

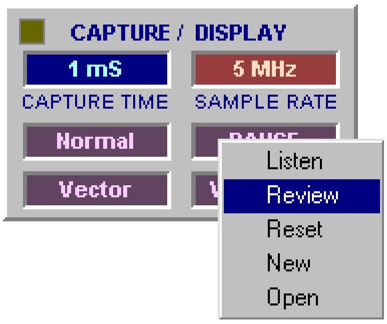
DSO DATA RECORDER

The data recorder saves captured waveforms and logic to DDR files

The BitScope DSO includes an integrated "DSO Data Recorder" (DDR).

DDR records selected analog and logic channels to an internal buffer for immediate playback, review and comparison with newly captured data.

Most importantly it also allows captured data to be recorded to **DDR Files** which may be reopened by the DSO for replay or analyzed with third party numerical analysis tools such as a spreadsheet or software such as MatLab.



DDR operation is integrated with the main DSO controls so other than choosing when to record and which file to record to you should not need to pay it much attention.

DDR commands are available via a pop-up menu selected by right-clicking the **DDR status**.

The example to the left shows DDR commands available when paused while recording data to the internal buffer.

Other commands are available in other modes such as when recording to a file or replaying captured data.

The selection of which channels to record is automatic; if the channels are visible on the main display they will be recorded by DDR when **Record** or **Listen** commands are selected and capture is commenced.

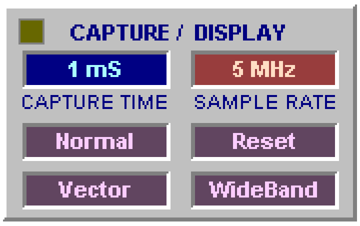
Likewise for replay; if the DDR file or buffer has data available for the channels selected for the main display they will be replayed when **Replay** or **Review** commands are selected.

The recorder's "transport controls", that is the controls that start it actually recording or replaying, are the same as those used by the DSO itself; the timebase **TRACE** and **REPEAT**.

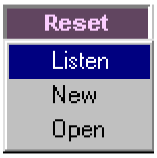
Basic Operation (1)

DDR exists in one of four states: **RESET**, **PAUSED**, **RECORDING** or **REPLAYING**.

The RECORDING and REPLAYING states have two "alter egos": **LISTENING** and **REVIEWING**.



These are the same as RECORDING and REPLAYING except that no DDR file needs to be specified and you can flip between them at will to compare new data with previously captured data.

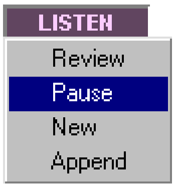


When the DSO is first started DDR is RESET.

This is indicated with the word **Reset** appearing in the DDR status as shown on the right and in this state three DDR commands are available; **Listen**, **New** and **Open**, as shown on the left.

Of these, **Listen** is the default and will be selected automatically if you left-click the DDR status.

Listen mode



Clicking the DDR status or selecting Listen from the menu puts DDR into listening mode as indicated by the word **Listen** which replaces Reset in the DDR status.

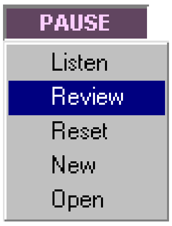
The next time TRACE or REPEAT is selected, the captured waveform and logic data is recorded to a buffer as well as displayed and the DDR status shows the word **LISTEN** (ie, upper case) showing data has been recorded.

Listening mode records selected waveform and logic data to an internal circular buffer. It does not require a DDR file and is a convenient mode to have enabled just in case "something interesting" happens. If it does, the recorded data can then be **saved to a file**.

Because the listen buffer is finite and circular previously captured data frames will eventually be overwritten with new ones unless you save them to a DDR file.

Pause mode

While listening (or recording, reviewing or replaying) clicking the DDR status or selecting **Pause** from the menu puts DDR into pause mode.



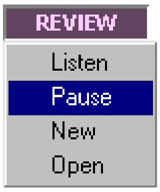
Pause mode stops DDR from listening (or recording, reviewing or replaying).

Clicking TRACE (or REPEAT) operates the DSO as usual but has no effect on DDR which remains idle but data previously recorded (or loaded from a file) remains loaded and available.

When paused you can continue to listen to more data by selecting **Listen** from the menu or review what you have already recorded by selecting **Review**.

Review mode

Review means to replay previously recorded data from the DDR frame buffer. When paused, left-clicking the DDR status selects Review mode (indicated by the word REVIEW). Clicking it again reselects PAUSE.

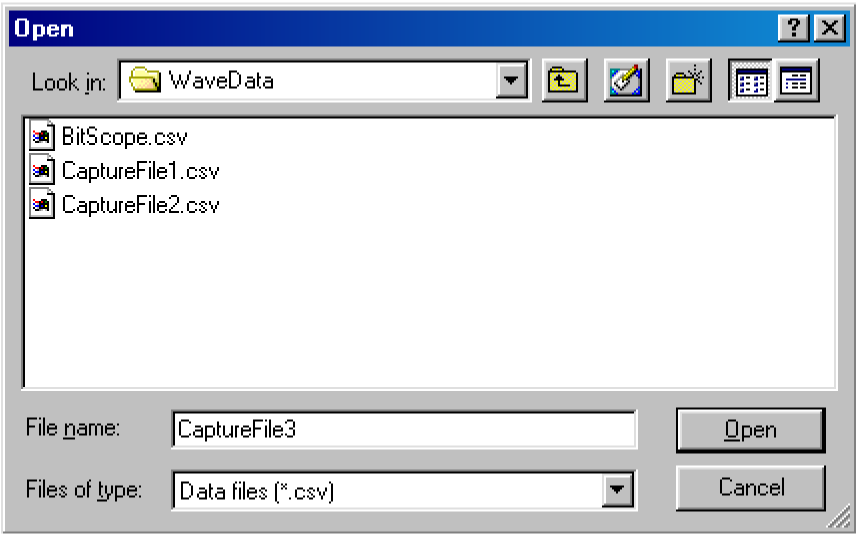


In this way it is simply a matter of repeatedly clicking the DDR status to toggle between viewing previously recorded data replayed by DDR and live data newly captured by the DSO.

File Operations (2)

Listening and reviewing analog waveforms and logic data is very useful in its own right but the main purpose of DDR is to record captured waveforms to **data files** for export to other analysis tools or for replay via another BitScope DSO.

When DDR is PAUSED, REVIEWING or LISTENING and has buffered data available, these data may be saved to a new file by selecting **New** from the menu. This opens a standard file dialog:



Look in: WaveData

BitScope.csv
CaptureFile1.csv
CaptureFile2.csv

File name: CaptureFile3

Files of type: Data files (*.csv)

By default the data is saved in the BitScope WaveData directory to a file named as you choose. If a file of that name already exists DDR will ask if you want to overwrite it.

Alternatively if you want to *append* the captured data to an existing file choose **Append** from the DDR menu instead of New. If an existing file is selected the new data will not overwrite data already existing in that file. If no file of that name exists one will be created.

The standard file name extension for DDR files is **.CSV** but you can choose a different extension if you prefer. Note that CSV is used for a reason; DDR files are ASCII encoded as CSV formatted data files so they are (with some care) suitable for direct import into spreadsheets, databases and any other software tool capable of importing CSV files.

To open a previously recorded DDR file for replay, choose **Open** from the menu. The file must exist and be a suitable DDR formatted CSV file otherwise DDR will ignore the file.

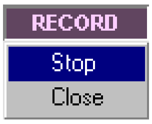
Data Recording (aka Data Logging) (3)

In addition to saving a finite duration of previously captured data stored in the buffer to a file, it is also possible to stream captured frames continuously to an open DDR file for as long as you have disk space available for the file as it grows in size.

Record mode

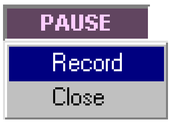
Record mode is effectively "listening directly to a DDR file" except that it is not possible to review recorded data until the file is closed and reopened for replay.

It is however possible to pause a recording (by left-clicking status or selecting **Stop** from the menu) and restart it without closing the file so you can make DSO setup changes without recording the interim junk data.



Record Pause mode

Record pause mode is similar to Pause mode but allows only the resumption of recording or closing of the DDR file. The default is to resume recording so toggling between recording and paused modes is simply a matter of clicking DDR status.



When a recording is closed, all remaining data in the buffer is flushed to disk to ensure no data is lost.

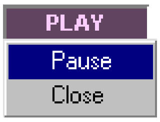
Data Replay (4)

One of the most useful features of DDR in collaborative or educational environments is its ability to replay previously recorded data to the display, even if no BitScope is connected to the PC.

With this feature it is possible to record reference waveforms and/or logic traces and distribute them to colleagues to compare with live data collected with other BitScopes. It is also possible to synthesize DDR files directly with a spreadsheet for replay via the DSO.

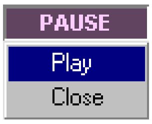
Replay mode

Replay mode is effectively "reviewing data directly from a DDR file" except that it is not possible to record more data until the replay file is closed and a new file opened for recording. It is however possible to pause replay (by left-clicking status or selecting **Pause** from the menu).



Replay Pause mode

Replay pause mode is similar to Pause mode but allows only the resumption of replay or closing of the DDR file. The default is to resume replay so toggling between the replay of a data file and display of live capture data is simply a matter of clicking DDR status.

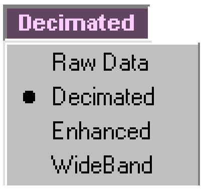


Data Format and Size (5)

DDR records whatever channels and data formats you select for display.

This ensures it can open and display a file without information loss.

However if your intention is to export data to another application you may wish to record the raw data (as captured by BitScope itself) or you may need to limit the amount data recorded per frame so it can be **imported to Excel**.



Use the **data mode** to choose the format of the data to be captured (and displayed). In particular choose **Raw Data** to record the raw data captured by BitScope but choose **Decimated** or **Enhanced** if you need a more compact encoding format.

Similarly, to maximize the amount of data recorded per frame in data modes other than Raw Data choose the **Dot/Vector** or **Dots** **display mode**. These modes have twice the display (and recording) resolution of the alternative **Vector** display mode.

File Format and Encoding (6)

DDR files are simply lists of data records encoded as ASCII values recorded to files in CSV format suitable for import into a spreadsheet or database (so long as certain conditions are met).

When recording, each data frame for every selected channel (both analog and logic) is appended the DDR file as a discrete CSV data record. This makes continuous data logging possible.

Metadata definitions

In addition to the capture data itself each record includes the following meta data:

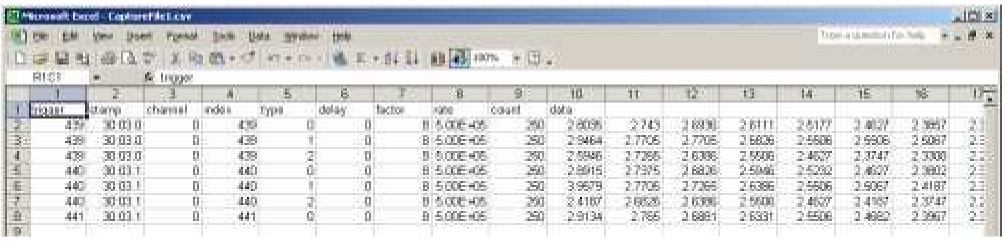
- **Trigger**: a sequence number identifying the trigger event that produced the record.
- **Stamp**: a "time of day" time stamp recording when the record was captured.
- **Channel**: the index of the channel that captured the data.
- **Index**: an index identifying all records belonging to the same frame.
- **Type**: the capture data type (analog, logic, decimated, filtered etc).
- **Delay**: the post trigger delay applied to the capture (if any).
- **Factor**: the decimation/oversample factor applied to the capture (if any).
- **Rate**: the sample rate at which the data was captured.
- **Count**: the number of samples captured in this record.

As all fields are ASCII encoded it is possible to open a DDR file with a simple text editor, word processor or spreadsheet to examine it.

Using a Spreadsheet

While CSV files are standard for spreadsheets, many (including Microsoft Excel) do not take kindly to being given too much data to load but using DDR it is very easy to produce vast quantities of data so there's a potential problem.

Specifically, if a CSV row contains more than 256 fields (ie, columns) Excel will refuse to load it. Similarly if it has more than 65k rows it will truncate it.


Microsoft Excel - CaptureFile1.csv
File Edit View Insert Format Tools Data Window Help
Type a question for help
100%
R1C1
trigger

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	trigger	stamp	channel	index	type	delay	factor	rate	count	data							
2	439	30:03.0	0	439	0	0	8	5.00E+05	250	2.8035	2.743	2.8030	2.6111	2.6177	2.4027	2.3957	2.
3	439	30:03.0	0	439	1	0	8	5.00E+05	250	2.9464	2.7705	2.7705	2.5826	2.5506	2.5506	2.5087	2.
4	439	30:03.0	0	439	2	0	8	5.00E+05	250	2.5946	2.7286	2.6386	2.5506	2.4627	2.3747	2.3308	2.
5	440	30:03.1	0	440	0	0	8	5.00E+05	250	2.8915	2.7375	2.6826	2.5946	2.5292	2.4027	2.3802	2.
6	440	30:03.1	0	440	1	0	8	5.00E+05	250	3.9579	2.7705	2.7265	2.6386	2.5506	2.5067	2.4187	2.
7	440	30:03.1	0	440	2	0	8	5.00E+05	250	2.4187	2.6526	2.6386	2.5600	2.4627	2.4187	2.3747	2.
8	441	30:03.1	0	441	0	0	8	5.00E+05	250	2.9134	2.765	2.6881	2.6331	2.5506	2.4882	2.3967	2.
9																	

If you're using DDR to produce files for import into Excel and similar spreadsheets you will need to ensure that the data records you produce are shorted than 256 fields and that you don't attempt to load more than 65k of them.

Because DDR is designed to avoid information loss (so you can reload a file and play it) it records as many samples per frame as the display requires to display the waveforms correctly. Further if you record raw data you potentially get *every* sample captured by BitScope (which can be more than 130k samples per frame per channel).

Clearly most raw data DDR files will be *big* and completely overwhelm most spreadsheets !

To use DDR files with Excel you must configure DDR to record fewer samples per frame. This will produce files which may be replayed via DDR but there will be some information loss so replayed waveforms will not be as precise as the originals and spectra produced from replayed data may be lower resolution.

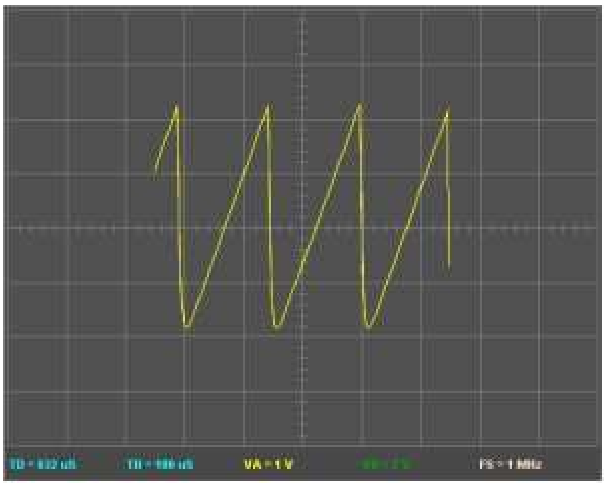
Waveform Intuitive Display Engine (WIDE) (7)

DDR was added to the DSO with another enhancement called *Waveform Intuitive Display Engine* (WIDE). WIDE replaces the old display engine to enable replay of DDR waveforms.

Unlike live capture, replaying previously recorded waveforms means you do not control how the waveform is triggered; instead this information is embedded in the recording.

Consequently if you are comparing new waveforms with previously recorded ones the trigger setup, waveform offset and delay may be different. However in most cases you will be interested to see both waveforms referenced to the same relative time (ie, a common trigger point).

This is what WIDE does; it displays waveforms relative to the trigger, timebase, waveform zoom and offset you have set up given the waveform data that's actually available for replay.



DDR waveform replay (zoomed out by 2)

Truncated Display

In practice this means is that replayed waveforms may be longer or shorter than the current display duration, or they may be offset to one side or the other (or not shown at all).

This screenshot shows one example. In this case the waveform was recorded at 50uS/Div but replayed at 100uS/Div.

As a consequence, data before the left edge and after the right edge of the display *as recorded* is not shown upon replay.

This is because it was not recorded in the first place.

DDR records whatever is displayed on screen at the time of the recording only.

Full Capture Recording

It is however easy to have DDR record as much data as BitScope can capture regardless of the displayed time duration to be used upon replay; simply choose the widest zoom and slowest timebase required to capture all the data of interest when recording.

If you need to maintain the highest time resolution (ie, sample rate) as well, choose the **largest buffer** size you can and capture **Raw Data**. Be warned though; doing this can produce very large DDR files and they will certainly not be suitable for import into a spreadsheet.

Variable Frame Rates

Another benefit of the WIDE engine, especially when used with DDR recordings is that you can change the rate at which multi-frame captures are replayed. If for example you recorded 200 frames over 24 hours you can replay them at whatever frame rate you like to make looking at frame-to-frame differences or transient features easier.

A related feature is that if you are using a BitScope via a slow link (eg, a remote Network BitScope via a dial-up link) but choose a fast frame rate, WIDE ensures that already displayed (ie cached) waveform data is redisplayed immediately upon a relevant parameter change (such as timebase or waveform offset) and new data is then appended dynamically when it finally arrives.