SJL
Instruments

GigaWave™ 6000-Series Digital Sampling Oscilloscope

User Manual · Revision H17 · April 20, 2024

Contents

Changelog

H17 (2024-04-20): Update documentation for automatic display of sampling timescale in power-user mode. Document enable/disable option for tooltip help system. Update documentation for import/export of multivalued signals (raw CDF data).

H16 (2024-04-04): Add User Manual changelog. Document serial protocol for device firmware update. Update FFT documentation. Expand documentation for self-test (`*TST?`) command.

H15 (2024-03-20): Update serial interface documentation for firmware v14. Add software documentation for fast mode. Add guidance on improving sweep speed. Add documentation for displaying timebase in distance units.

1 Specifications

1.1 Absolute Maximum Ratings

Exceeding these specifications, even briefly, may cause permanent damage to the device.

Note: Signals larger than the absolute maximum range can be safely measured using an external SMA attenuator.

Parameter	Rating
Input Voltage (any channel)	–1.2 V to +1.2 V
Operating Temperature	–20°C to +50°C
Storage Temperature	–60°C to +85°C

Warning! The scope inputs are ESD-sensitive. Use the included SMA shorts to protect unused channels. An ESD-damaged channel will fail open (high-impedance).

1.2 Physical Specifications

Parameter	GigaWave™ 6200	GigaWave™ 6400	GigaWave™ 6800
Dimensions (L×W×H)	3.0×2.7×0.5"	4.0×2.7×0.5"	7.0×2.7×0.5"
	76×69×13mm	102×69×13mm	178×69×13mm
Weight	0.25 lb	0.33 lb	0.62 lb
	115 g	148 g	280 g
Connectivity		USB 3.0 Type-C	
USB Current Draw (5 VDC)	700 mA	850 mA	1.5 A
Power Draw	3.5 W	4.3 W	7.5 W
Input Interface	2× SMA Female	4× SMA Female	8× SMA Female
Chassis		Machined Aluminum 6061	

1.3 General Specifications

Specified at an ambient temperature of T_A = 25 °C.

Parameter	Minimum	Typical	Maximum
Input Impedance (1 GHz)	47 Ω	50 Ω	53 Ω
Input Coupling	—	DC	—
Input Voltage (any channel)	–950 mV	—	+950 mV
Trigger Source	—	Channel 1 Edge	—
Analog Bandwidth (–3 dB)	6 GHz	8 GHz	—

1.4   Timing

Performance guaranteed only after a 5-minute warmup period.

Parameter	Minimum	Typical	Maximum
Temporal Resolution	—	0.3 ps	1 ps
Timebase Range (Post-Trigger)	11 ns*	—	20 μs
Timebase Accuracy			
Scaling (Linearity)	—	15 ppm	30 ppm
Absolute (Global Shift)	—	20 ps	50 ps
Inter-Channel Skew	—	5 ps	15 ps

*Can be decreased to 8 ns or earlier in power-user mode (Section 3.8.5).
Channel 1 will experience increased distortion between 8 and 11 ns.

1.5   DC Performance

After a 5-minute warmup period, the RMS noise performance will meet or exceed the typical specification. The maximum specification applies immediately after power-on.

Parameter	Minimum	Typical	Maximum
Vertical Resolution	—	12-bit	—
Offset Error (Input Shorted)	—	1 mV	4 mV
RMS Noise	—	1 mV	3 mV

1.6   AC Performance

Note: The maximum THD and minimum ENOB specifications are limited by increased distortion in the 11-12 ns region. The typical specification is achieved at 12 ns and later.

Parameter	Minimum	Typical	Maximum
Rise/Fall Time (20–80%)	—	40 ps	50 ps
Total Harmonic Distortion (Full-Scale)			
100 MHz Sine Wave	—	−55 dB	−50 dB
1 GHz Sine Wave	—	−40 dB	−30 dB
Effective Number of Bits (Full-Scale)			
100 MHz Sine Wave	8.0	8.5	—
1 GHz Sine Wave	5.0	5.5	—
Trigger			
Bandwidth (Full Scale Edge)	6 GHz	7 GHz	—
Minimum Edge Amplitude[†]	40 mV$_{PP}$	200 mV$_{PP}$	—
Trigger Level	−950 mV	—	+950 mV
Trigger Level Resolution	—	12-bit	—
Holdoff	40 ns	—	21 μs
Intrinsic RMS Jitter*	—	4 ps	8 ps
Slew Rate Dispersion	—	5 ps	10 ps
Real-Time Trigger Rate (40 ns Holdoff)	—	20 Mtrig/s	25 Mtrig/s
Equivalent Sample Rate*	—	4 Msa/s	—

*See Section 2.1.2 (CDF Sampling) for details.

[†]See Section 1.6.2 (Trigger Sensitivity).

1.6.1   Note: THD and ENOB

Reported THD and ENOB on the device-specific calibration sheet are extracted from a time-domain regression with the following timebase settings:

100 MHz: 300 ns position, 500 ns holdoff @ 10 ns/div, 32 points/div.

1 GHz: 50 ns position, 100 ns holdoff @ 1 ns/div, 32 points/div.

The regression model used is

$$V = V_0 + \sum_{n=1}^{9} V_n \sin(2\pi nft + \phi_n),$$

where V_n, f, and ϕ_n are free parameters. The residuals are verified to be free of harmonic content, and are used to calculate the noise power.

1.6.2 Note: Trigger Sensitivity

The minimum signal amplitude for a reliable trigger depends on the vertical scale of Channel 1.

The highest trigger sensitivity is achieved with Channel 1 set to 5 mV/div, and the offset adjusted to vertically center the trigger level. In this configuration, signals as small as 40 mV$_{PP}$ can be used.

In Power-User mode (Section 3.8.5), the CDF search range for Channel 1 can be manually specified, allowing for finer control over the trigger sensitivity. For very small signals (below 15 mV$_{PP}$), the trigger level may need to be adjusted up or down by a few millivolts to obtain a stable trigger. The precise minimum amplitude required for a trigger is frequency-dependent and subject to process variation.

1.7 PC Requirements

All GigaWave™ oscilloscopes are powered by the host computer's USB port. GigaWave™ 6200 and 6400 models require a USB 3.0 port (900 mA @ 5 VDC). The GigaWave™ 6800 model requires a USB-C port capable of supplying 1.5 A (such as a Thunderbolt port or dedicated charging port).

The GigaWave™ software will run on any 64-bit installation of Windows 10, macOS Big Sur, or Ubuntu 20.04 (or newer). See Section 3.1 for installation instructions.

The application requires 200 MB of disk space (400 MB on macOS), and at least 512 MB of RAM.

Please check that your host computer meets these requirements prior to purchase.

1.8 Accessories

All GigaWave™ oscilloscopes come with the following standard accessories:

- A 3' USB-C to USB-C power cable,

- A USB-A to USB-C adapter,

- A Torx T10 hex key (for opening the case), and

- A calibration sheet with measured bandwidth, risetime, trigger jitter, THD, and ENOB for your specific unit. (See an example calibration here.)

If any accessories are missing, please contact us for a free replacement.

If using a USB-A port, please verify that the port supports USB 3.0 power levels (900 mA @ 5 VDC).

The GigaWave™ can be run over longer USB cables or USB extension cables without issue. For very long cable runs, a powered USB hub may be necessary.

2 Theory of Operation

2.1 Architecture

2.1.1 Sequential Sampling Oscilloscopes

The GigaWave™ is a **sequential sampling oscilloscope**. This type of oscilloscope works by sampling the input signal(s) a fixed time $\triangle t$ after a trigger occurs. The post-trigger delay $\triangle t$ may then be varied across a range of values to build up a waveform.

Compared to a real-time oscilloscope, this class of instruments places two main requirements on the input signal:

1. The signal of interest must be **repetitive** (but not necessarily periodic). Each time the scope is triggered, a small amount of information is gathered about the input signal. Many (thousands to millions) of triggers are required to build up a viewable waveform.

   If the signal is not identical on each repetition, the GigaWave™ can display an intensity-graded representation of the waveform, but cannot give insight on single-shot behavior.

2. The instrument can only view events that occur **after the trigger**. In particular, you cannot view the edge that is being triggered on. The usual workarounds for this are:

   (a) Tapping into a **pretrigger** (an event that occurs a fixed time before the signal), and sending the signal of interest into another channel,

   (b) Looking at the next occurrence of the signal after the trigger (for periodic signals only), or

   (c) Splitting off part of the signal into a coaxial cable to generate an **analog delay**. (This can distort the signal. In some cases, the effect of the cable can be de-embedded; see Section 3.7.)

In many cases, it is possible to create a pretrigger by delaying the digital trigger for the device under test. For this purpose, we recommend using a 12 foot (3 m) coaxial cable.

If the signal of interest does not meet the above requirements, the GigaWave™ is not a suitable instrument. We would instead recommend a random sampling oscilloscope or a real-time oscilloscope.

2.1.2 CDF Sampling

In a traditional sequential sampling scope, the signal of interest is sampled at a fixed time $\triangle t$ after the trigger event. The analog sample is then digitized by an analog-to-digital converter (ADC). This procedure is repeated several times to build up the effective probability density function (PDF) of the signal as a function of voltage V, denoted $P(V; \triangle t)$.

In contrast, GigaWave™ scopes obtain the same PDF by directly measuring the corresponding **cumulative distribution function (CDF)**, denoted $F(V; \triangle t)$. The CDF is then numerically differentiated to reconstruct the PDF of the sampled signal.

In practice, this CDF sampling architecture has three primary advantages:

1. **Reduced cost.** The front-end of the instrument can be significantly simplified by using CDF sampling. This greatly reduces the cost of the instrument, allowing the GigaWave™ to offer 6 GHz bandwidth at an industry-leading price.

2. **Increased trigger rate.** The hardware required to measure the CDF achieves real-time trigger rates up to 25 Mtriggers/second, limited only by the minimum trigger holdoff of 40 ns. In contrast, traditional sampling scopes generally operate at rates of 100k-1M triggers/second.

3. **Information efficiency.** In general, the PDF has significant structure. From an information-theoretic perspective, the traditional sampling method reconstructs this structure in an inefficient way.

   For instance, consider taking 100 samples with a 12-bit ADC. If the signal is single-valued (with some noise) at the fixed delay $\triangle t$, the high-order bits from the ADC will be redundant. Thus one will obtain much less than 1200 useful bits of information about $P(V; \triangle t)$. More generally, this holds true as long as the signal of interest has any structure relative to the trigger point.

   In contrast, GigaWave™ scopes can allocate triggers to measure the CDF in an information-theoretically optimal manner. As data are collected, the scope prioritizes sampling of the most interesting portions of $F(V; \triangle t)$. In practice, we measure 5–10$\times$ improvement in bitstream efficiency relative to a traditional ADC across a range of real-world signals.

Effects 2 and 3 compound to allow fast acquisition of complex signals with high repetition rates, such as eye diagrams or modulated waveforms.

The CDF sampling technique will be slower for signals with infrequent trigger opportunities (500 ktrig/second or lower). Unlike an ADC-based sampler, the CDF hardware extracts one bit of information per channel, per trigger.

Therefore, if the trigger rate is inherently limited by the signal source, then the wall-clock acquisition speed of the GigaWave™ scope may be 2–4$\times$ slower than an equivalent traditional sampling scope, depending on the level of structure in the signal (point #3 above). In the worst-case, when the signal has no structure at all, the acquisition may be 12$\times$ slower than a 12-bit traditional sampling scope.

2.2 Choosing CDF Settings

2.2.1 Definitions

During operation, the scope will park at a fixed post-trigger delay Δt and measure the CDF $F(V; \Delta t)$ at various voltages V. This value represents the probability that the signal at time Δt is less than the query voltage V.

The precise sequence of query voltages V is chosen automatically to extract as much information as possible from the CDF (in the sense of Fisher information).

The **number of voltages** in this sequence is denoted K and defaults to 40. This parameter will determine the **complexity** of features that can be resolved.

At each voltage V, the **number of triggers** N used to measure $F(V; \Delta t)$ will lie between user-definable parameters $N_{\min}$ and $N_{\max}$. These parameters will determine the **accuracy** of the results.

Detailed guidelines for choosing these parameters are given in the following sections.

2.2.2 Acquisition Speed

The theoretical time to acquire the CDF for one post-trigger delay Δt can be computed as

$$\tau_{\mathsf{CDF}} = K N_{\min} / f_{\mathsf{trig}},$$

where f_{trig} is the trigger frequency. The total time required for one sweep is τ_{CDF} times the number of points in the timebase.

For example, at a timebase resolution of 32 points/division, a total of $12 \cdot 32 = 384$ CDFs must be acquired. With the default settings $K = 40$ and $N_{\min} = 4096$, at a trigger rate of $f_{\mathsf{trig}} = 10$ Mtrig/s, each CDF requires 16.4 ms to be acquired. This corresponds to a theoretical sweep time of 6.3 seconds.

Increasing K and $N_{\min}$ will improve the accuracy of the result, while decreasing them will result in faster acquisition. There is therefore a **tradeoff between accuracy and speed**. We recommend using low values for K and $N_{\min}$ for setting up the timebase and vertical bounds, then increasing K and $N_{\min}$ for final acquisition.

The actual sweep time will be larger due to communication overhead, and will depend slightly on your computer's USB hardware.

Note: Firmware versions 13 and below have a minimum communication overhead of 15 ms per CDF.

2.2.3 General Guidelines: K (Samples per CDF)

While the scope has 12-bit vertical resolution, the number K of sampled voltages determines the level of detail for resolving complex features.

If the signal is a single-valued trace, then low values of $K \sim 20$ are acceptable, and doing so will speed up acquisition. At each delay Δt, all the sampled voltages will be clustered around the signal voltage, with a spread whose scale corresponds to the RMS noise in the signal. This will accurately resolve the noise profile and mean value at each delay.

If the signal is multivalued or contains complex modulations, then larger K is necessary to resolve details. If K is set too small, there will be visible steps or blocks in the intensity grading (shown below), corresponding to the regions between the sampled voltage levels.
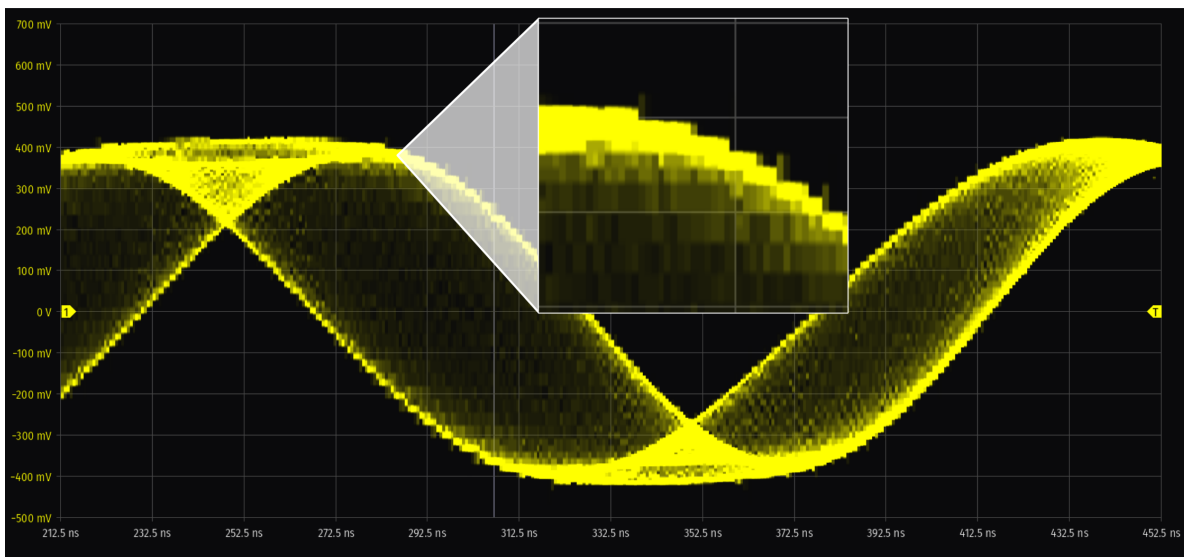
The examples on the next page show a PAM4 eye diagram with $N_{\mathrm{min}} = 20000$. From top to bottom, we have set $K = 20$, $K = 40$, and $K = 100$. The lower values $K = 20$ and $K = 40$ are not sufficient to resolve the complex behavior during the transition region.

For two-level eye diagrams, the default value $K = 40$ is generally sufficient, although larger values can be used to resolve finer details.

Note: We do not recommend the GigaWave for more complex modulations, such as PAM8 or QAM. The required K scales with the number of transitions between levels, so a PAM16 eye diagram with $16^2 = 256$ transitions would require a $16\times$ larger K than a PAM4 eye diagram with $4^2 = 16$ transitions. This will slow down acquisition to impractical levels.

Note: In firmware revisions 13 and below, setting K larger than 100 will fallback to direct software sampling of each voltage level. This will dramatically slow down acquisition.

SJL Instruments
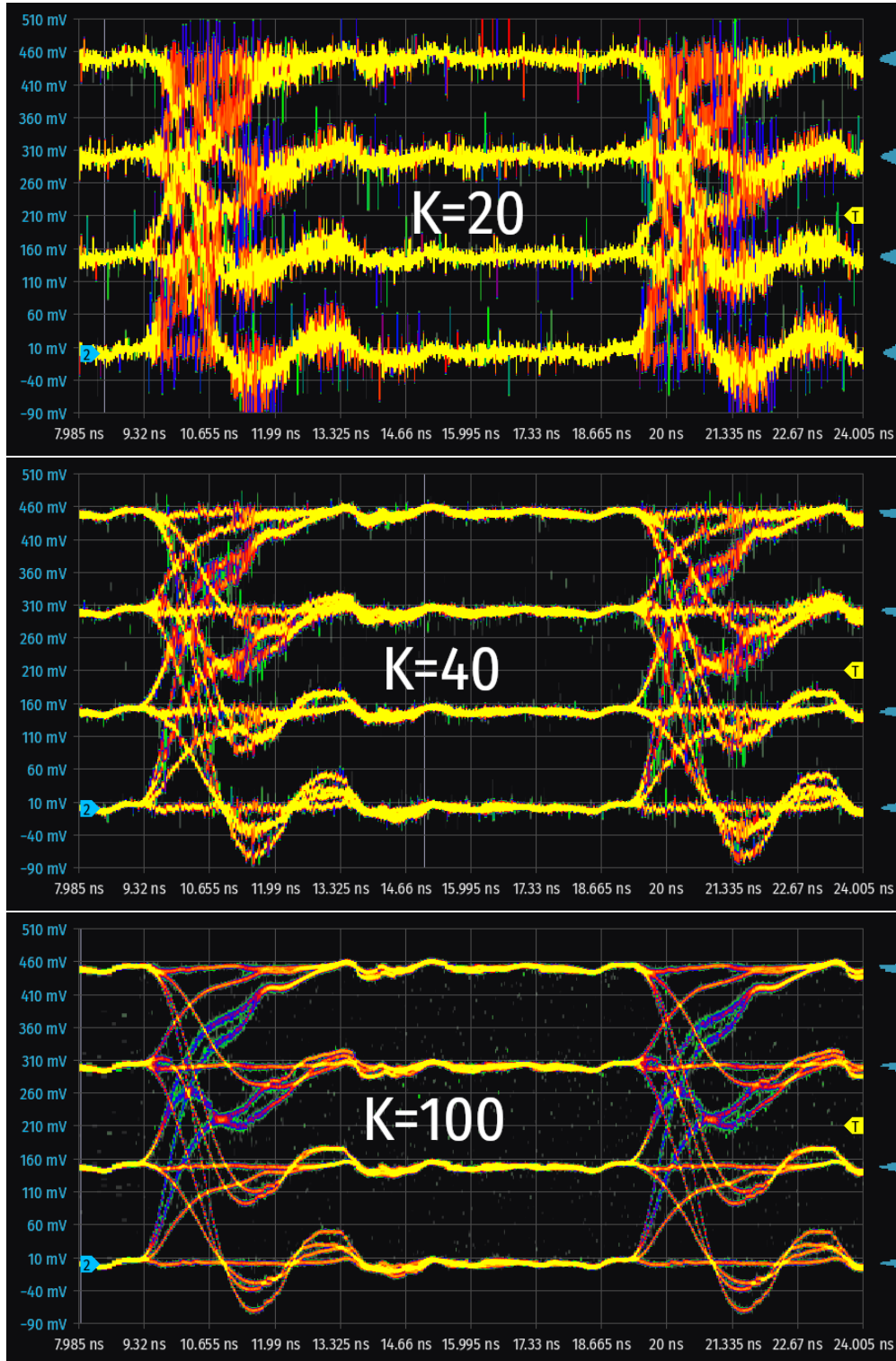
2.2.4 General Guidelines: $N_{\text{min}}, N_{\text{max}}$ (Triggers per CDF Sample)

The parameters $N_{\text{min}}, N_{\text{max}}$ constrain the number of triggers N used to measure the CDF at each query voltage. More triggers will be allocated to more information-dense regions of the CDF. (See Section 2.2.5 for details).

In practice, we recommend setting $N_{\text{max}} = \infty$ and only changing N_{min}, except when the trigger source is infrequent (<500 ktrig/s). For guidance on infrequent trigger sources, see Section 2.2.6.

The value of N_{min} determines the level of statistical noise in the displayed signal. If this noise level is too high (in particular with eye diagrams), speckles may appear within the open eye.

The examples below show a two-level eye diagram with $N_{\text{min}} = 1000$ (left) and $N_{\text{min}} = 10000$ (right). We use the default $K = 40$. For applications where minimizing this effect is critical, or if setting $N_{\text{min}} > 20000$, see Section 2.2.8.
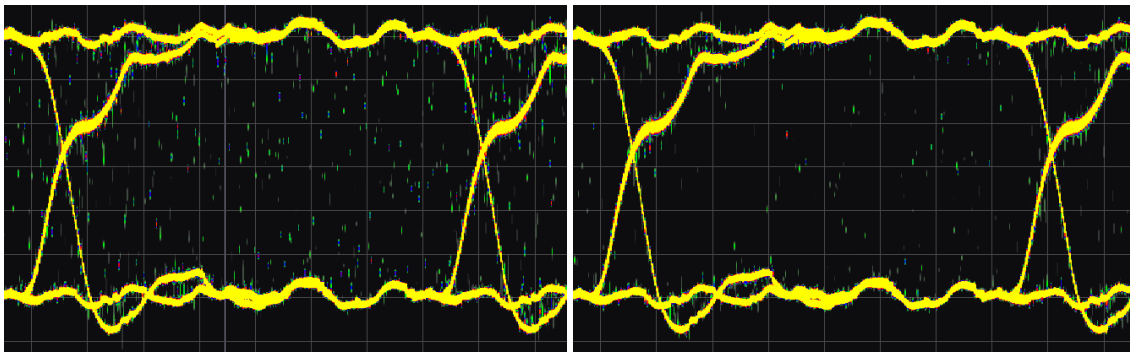
The default parameters are $N_{\text{min}} = 4096$ and $N_{\text{max}} = \infty$ for 40 ns holdoff, with N_{min} decreasing as holdoff increases to maintain a sampling timescale $\tau_0 \sim 160$ µs (see Section 2.2.7).

Note: In firmware revisions 13 and below, setting N_{min} larger than 30000 will fallback to direct software sampling of each voltage level. This will dramatically slow down acquisition.

2.2.5 Advanced Guidelines: N_{min}, N_{max} (Triggers per CDF Sample)

In detail, sampling is complete when the number of triggers N exceeds N_{max}, or when the statistical uncertainty on $F(V; \Delta t)$ falls below a threshold $\epsilon = 0.01$, whichever occurs first. For a given number of triggers N, the statistical uncertainty on F is given by

$$\delta F = \sqrt{\frac{F\,(1-F)}{N}}. \tag{*}$$

Thus the number of samples needed to reach the target precision ϵ is given by

$$N_{\text{target}} = F\,(1-F)/\epsilon^2.$$

As expected, fewer samples are needed to reach the target precision ϵ when the CDF is close to 0 or 1 (uninteresting regions).

It can be shown that $N_{\text{target}} \leq 1/(4\epsilon^2) = 2500$ for $\epsilon = 0.01$. Therefore, setting N_{max} larger than 2500 has no effect.

We do not recommend changing ϵ, as this may lead to inaccurate display of the PDF. This parameter can currently only be changed via the serial interface.

Note: For bit error ratio (BER) testing, the formula (*) shows that achieving half the BER requires 4 times more triggers ("square root scaling"). For this reason, we **do not recommend the GigaWave for BER testing.** This limitation can be resolved with a dual-comparator architecture - contact us for more information if interested.

2.2.6   Infrequent Trigger Sources

Decreasing N_{max} is necessary for signals with low trigger rates (<100 ktrig/s), as acquisition may timeout and the scope will enter "wait" mode.

Note that very low values of N_{max} and K may not provide enough information for accurate reconstruction of the PDF, causing occasional glitches or display artifacts. We recommend using the following minimum settings, or higher:

Trigger Rate f_{trig}	Minimum N_{min}	Minimum N_{max}	Minimum K
100 Hz – 3 kHz	15	15	15
3 kHz – 30 kHz	50	200	15
30 kHz – 100 kHz	200	500	20
>100 kHz	500	2000	20

### 2.2.7 Modulated Signals: CDF Stability and the Sampling Timescale

For proper reconstruction, the CDF must remain stable while it is being sampled. In detail, for a fixed post-trigger delay Δt and voltage V, the oscilloscope will spend a **sampling timescale** τ_0 measuring the CDF $F(V; \Delta t)$. The sampling timescale can be computed as

$$\tau_0 = N_{\text{min}}/f_{\text{trig}},$$

where N_{min} is the minimum number of triggers per CDF sample and f_{trig} is the trigger rate.

In **Power-User mode** (Section 3.8.5), this timescale is calculated and displayed automatically.

Signal variation (e.g. modulation) over timescales shorter than τ_0 will be represented properly on the intensity-graded display. Variation on timescales longer than τ_0 will create inconsistencies between the sampled CDF values $F(V; \Delta t)$ for different V, leading to possibly erratic results.
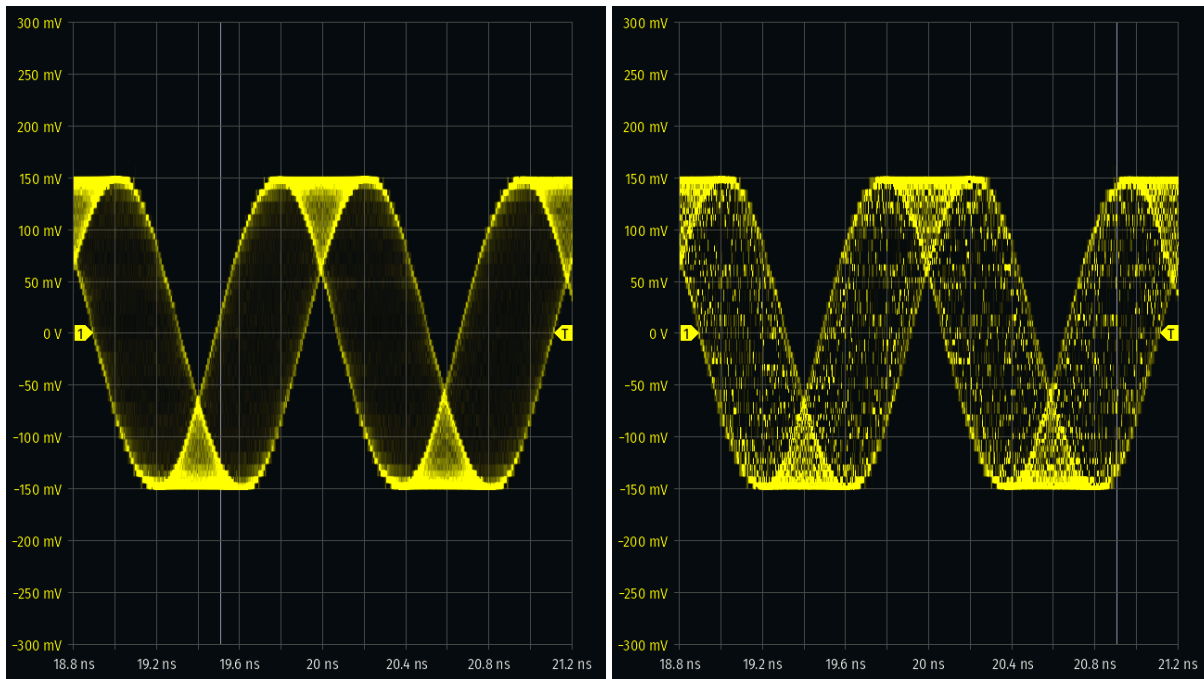
For example, the two traces above show an 840 MHz sine wave carrier with 1% FM modulation. Here we have set $N_{\text{min}} = 1000$, and we reach the maximum trigger rate of $f_{\text{trig}} = 25$ Mtrig/s. The sampling timescale τ_0 is therefore 40 μs.

- On the left, the FM modulation occurs at 100 kHz (period 10 μs), faster than τ_0. The displayed intensity-grading is accurate.

- On the right, the FM modulation occurs at 10 kHz (period 100 μs), slower than τ_0. While the general shape of the waveform is reconstructed correctly, the intensity-grading is not accurate.

If the modulation timescale is not known, we recommend using a large value for N_{min} (20000 or higher), although this will slow down acquisition.

2.2.8 Advanced Guidance: Reducing Eye Speckle

See Section 2.2.4 prior to reading this section.

A small portion of the CDF noise (and thus the resulting speckle) is not attributable to statistical noise. This can cause periodic patterns in the speckle density.

We first provide appropriate software settings for eliminating this effect, followed by an explanation of the source.

Settings: For optimal performance, the following settings should be used:

- Set all channels, including hidden channels, to -10 V offset and 5 mV/division, **except** the channel of interest and Channel 1.

- Set Channel 1 (if not the channel of interest) to 5 mV/division, clipping the trigger waveform. Adjust the Channel 1 vertical offset to center the trigger level on the display.

Root Cause: The root cause is subtle. Consider viewing an eye diagram on Channel 2, with a clock on Channel 1, and Channels 3-8 disconnected. During acquisition, the GigaWave changes the query voltage for all channels in parallel. Thus the comparator outputs on Channels 1 and 3-8 will toggle while Channel 2 is being acquired.

There is a small amount ($\sim$100 µV) of digital-to-analog crosstalk from these comparator outputs onto Channel 2. This has the effect of shifting the measured CDF by 10-100 ppm. This shift will cancel out during CDF to PDF differentiation if the other comparators have a constant output, but will lead to speckles if the other comparators are toggling.

The recommended settings above will cause Channel 1 and Channels 3-8 to always clip, forcing their comparator outputs to a constant at each post-trigger delay Δt. This eliminates the effect.

2.3 Timebase Jitter

A delay generator is used to control the post-trigger delay Δt during acquisition. Due to the inherent jitter of the delay generator, the apparent (measured) jitter of an edge will increase with Δt according to the formula

$$T_{\text{total}} = \sqrt{T_{\text{true}}^2 + T_{\text{trigger}}^2 + \alpha \Delta t},$$

where T_{true} is the inherent jitter of the edge, T_{trigger} is the trigger jitter, and $\alpha \approx 0.5 \text{ ps}^2/\text{ns}$ characterizes the random jitter of the delay generator. The device-specific parameters T_{trigger} and α are provided in the calibration certificate.

For jitter-sensitive applications, we therefore recommend keeping the signal of interest **within 100 ns** of the trigger.

3   Software

3.1   Installation

3.1.1   Windows

Download and extract the `.zip` file containing the GigaWave™ software. Run the **GigaWave.exe** executable. If a SmartScreen warning occurs, click on **More info** and then **Run anyway**.

There may be a 10 second delay on first startup as Windows Defender scans the executable.

Appropriate drivers are usually installed automatically within 30 seconds of plugging in the device. If no device is recognized, manually install the **CP2102x USB to UART Bridge VCP Driver** from Silicon Labs, then disconnect and reconnect the scope.

3.1.2   macOS (beta)

Download and install the **CP2102x USB to UART Bridge VCP Driver** from Silicon Labs. During driver installation, click **Open System Settings** and then **Allow** if a "System Extension Blocked" message appears.

After driver installation completes, download and run the GigaWave™ software.

On macOS, the scope **must be plugged in** prior to launching the software.

Note: The macOS software is currently in beta. The display may occasionally become unresponsive to trackpad movements. This can be fixed by performing a three-finger swipe to a different app and back.

3.1.3   Ubuntu Linux

Download and extract the `.zip` file containing the GigaWave™ software. Run `./gigawave` to start the software.

Linux kernel versions v2.6.12 and newer include the appropriate driver by default.

3.2   Overview and Controls

An overview of the software is shown below. The post-trigger delay Δt is swept from left to right, as shown by the vertical sweep indicator (white line). As the line is swept, the corresponding data is updated for all traces. One full sweep takes approximately 8 seconds with the default settings, corresponding to roughly 2 million samples.

Quick controls:

- Use the scroll wheel while hovering over the main display to **zoom the timebase** in and out. On touchscreen, pinch horizontally to zoom.

- Drag horizontally to shift the **timebase position**.

- Use the scroll wheel while hovering over a channel marker to adjust the **vertical scale**. On touchscreen, pinch vertically to adjust the scale of the focused channel.

- Channel markers and the trigger level can be dragged vertically.

- The trigger holdoff marker (if visible on timebase) can be dragged horizontally.

- To change the focused channel, hover over (or tap, on touchscreen) a channel marker or the settings box on the right-hand side of the screen.

3.2.1   Acquisition

This section controls the acquisition (start/stop) and displays the trigger rate. When stopped, no new data will be acquired, but all other functionality is available.
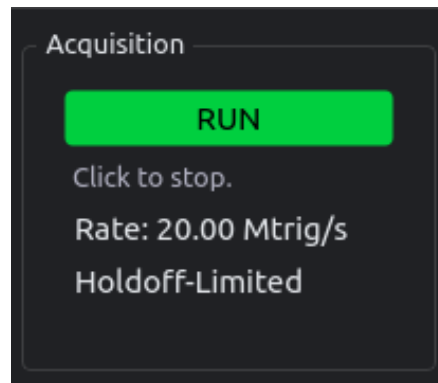

When triggering normally, the trigger rate may be limited by the source frequency (source-limited) or by the configured trigger holdoff (holdoff-limited). If the source period and the holdoff duration have a small common multiple, the trigger rate may be limited to a lower rate determined by the interaction of the two timescales.

When no trigger is present, the scope will trigger off its internal clock (100 Mtrig/s) until a valid trigger source returns.

Note that the displayed trigger rate does not include communication overhead; the effective real-time trigger rate may be lower. On older computers, the overall acquisition rate may be limited by CPU speed.

3.2.2   Trigger

This section controls the direction (rising or falling edge) of the trigger, as well as the trigger level. Only edge triggering on Channel 1 is available.
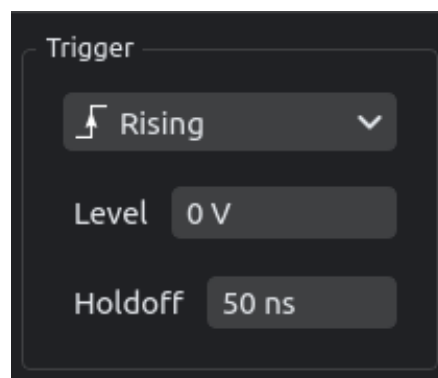
The trigger level ranges from −950 to +950 mV, and can be set with 12-bit resolution. The trigger level may also be changed by dragging the yellow **T** marker on the right side of the main display.

The trigger holdoff controls the minimum allowable delay between trigger events. The usable timebase starts 11 ns after the trigger and ends 25 ns before the trigger holdoff.

Burst events (e.g. serial data packets) can be observed by setting the trigger holdoff longer than the duration of each burst, but shorter than the gap between bursts.

3.2.3 Timebase

This section controls horizontal position, scale, and resolution of the main display.

These settings may also be changed by dragging the main display horizontally, and using the scroll wheel to zoom. On touchscreen devices, the scale can be changed by pinching horizontally.

3.2.4 Saving and Restoring the Workspace

The current workspace configuration can be saved to a file using **File → Save Settings**. You can restore a previously saved workspace using **File → Load Settings**.

To load a default workspace on startup, save the settings file as **default.json** in the `_internal` folder. The software will automatically search for a settings file with this name. (This feature is not currently available on macOS.)

Tip: The shortcut **Ctrl + S** saves the current workspace as **default.json**.

3.2.5 Built-In Help (Tooltips)

A tooltip will appear after hovering over any control for a few seconds. This provides a brief summary of what the control does, as well as a reference to the appropriate section of the manual (where applicable).
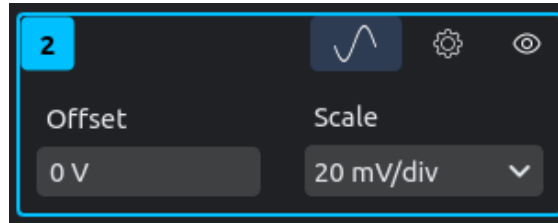
The tooltips can be globally disabled using **Options → Enable Tooltips**.

3.3 Channel Settings

3.3.1 Basic Settings

Tip: To quickly change the vertical scaling for a channel, hover over the channel marker and use the scroll wheel.

Settings for each channel are located on the right side of the window. Each channel may be selectively enabled or disabled using the eye icon.

The vertical scale on the main display corresponds to the currently focused channel. The focused channel can be changed by hovering over the corresponding channel settings panel, or the corresponding channel marker (located on the left edge of the main display).

The vertical offset of the channel can be changed by dragging the corresponding channel marker. The vertical scale can be changed by using the scroll wheel on the channel marker (desktop), or by vertical pinching (touchscreen, zooms the focused channel).

Tip: A custom vertical scale can be entered by selecting the **Custom...** entry from the dropdown.

3.3.2 Vector Trace

The sine wave icon on each channel enables or disables vector trace mode.

When vector trace mode is disabled, the full distribution of the measured signal is displayed with intensity grading.

When vector trace mode is enabled, the channel is drawn with one continuous line that follows the average of the measured sample distribution. The standard deviation will be shown as a band around the trace, indicating of the noise level.

This mode **must be enabled** for certain operations, such as the FFT, de-embedding, math channels, and all periodic measurements (frequency, duty cycle, etc.).
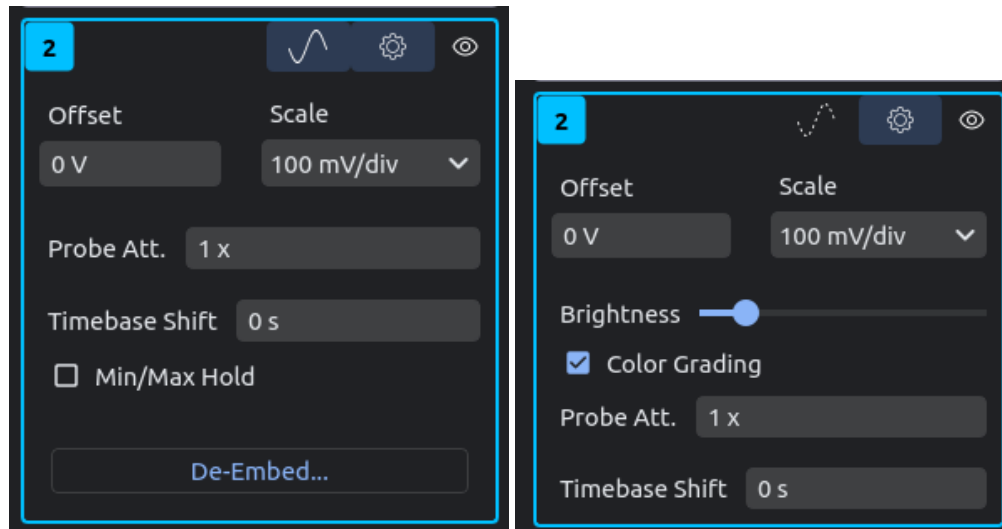
Enabling this mode when the signal is multivalued (e.g. an eye diagram) may lead to meaningless results.

3.3.3   Extended Settings

Clicking the gear icon shows and hides the extended settings panel.
Different settings are available depending on whether the channel has vector trace mode enabled (left) or disabled (right).

Settings available in **all modes**:

- The **probe attenuation** (can be specified in dB).

- A channel-specific **timebase shift**, which relabels the horizontal timebase. This can be used to compensate for coaxial cable delays.
  (The currently displayed shift is determined by the focused channel.)

Settings available only when vector trace mode is **enabled**:

- **De-embedding** of coaxial cables or other sources of distortion. See Section 3.7 for documentation.

- Checkbox to show or hide **min/max hold** traces for this channel.

Settings available only when vector trace mode is **disabled**:

- Slider to adjust the **brightness** of intensity-grading or color-grading.

- Checkbox to enable/disable **color grading**.

3.4   Analysis Tools

3.4.1   Cursors

Cursors are shown by default and follow the mouse. They may be visually hidden using the 'Show/Hide' checkbox, but all functionality remains available.

The voltage and time values at the cursor location are shown for the currently focused channel. The focused channel may be changed by hovering over the corresponding channel marker or channel settings panel.

To place a cursor, right click the display (desktop) or long tap (touchscreen). Cursor 2 may be placed after placing Cursor 1. After both cursors are placed, right clicking again clears the placed cursors.

A placed cursor is referenced to the corresponding X and Y values, and will therefore follow the screen as it is scaled or translated. Changing the focused channel will move the cursor to the corresponding voltage on the new vertical scale, allowing direct comparisons between different channels.

3.4.2   Histograms

Histograms can be enabled or disabled using **Options → Show Histograms**.

When vector trace mode is enabled (top), the histogram is computed assuming that the trace is continuous; i.e., filling in the gaps between samples.

When vector trace mode is disabled (bottom), the histogram is computed based on the raw CDF data. This may leave gaps in the histogram at low timebase resolutions.
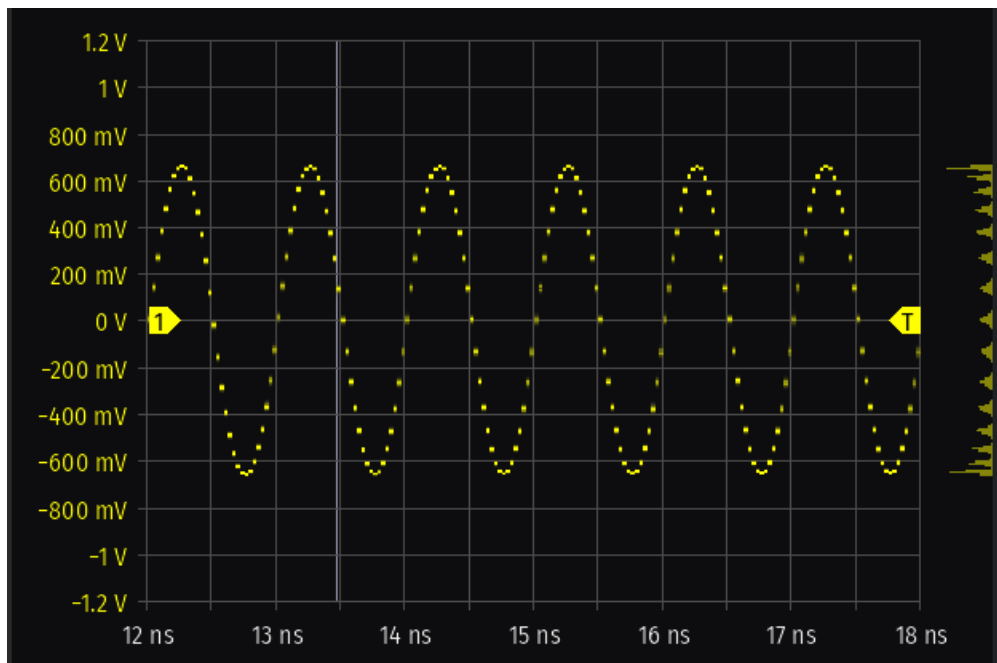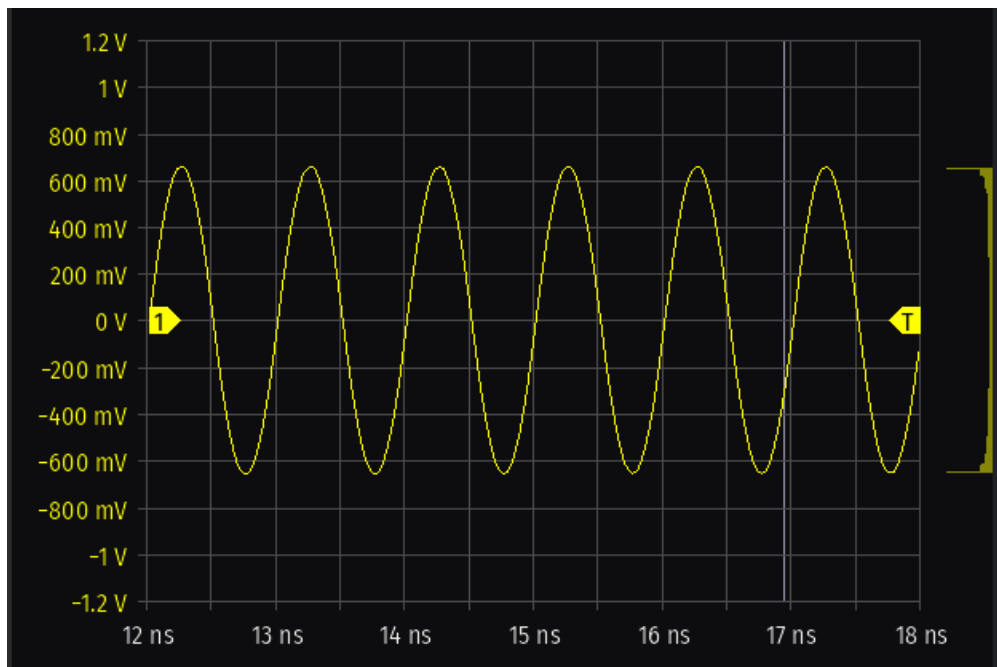
3.4.3 FFT

The FFT utility can be accessed using **Tools → FFT**. Multiple FFT windows can be open simultaneously.
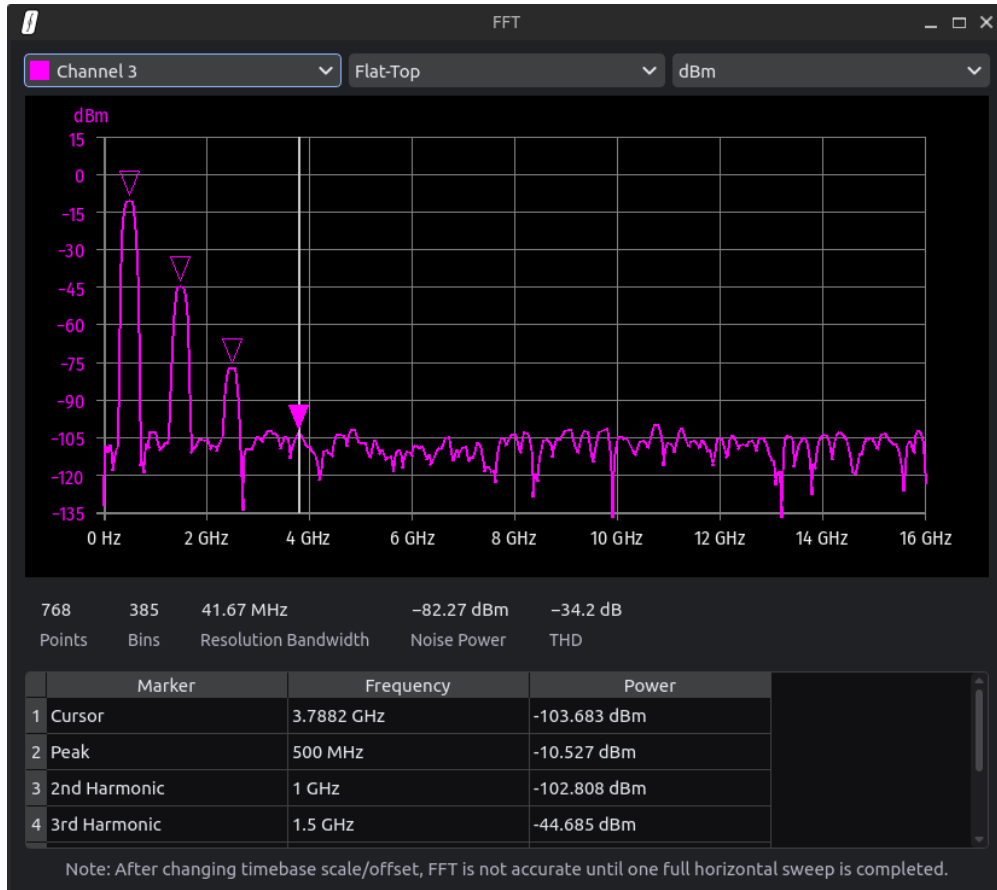
At the top of the window are settings for the channel, window type, and vertical units. Below the main FFT display are various useful statistics. At the bottom of the window are readouts for the interactive cursor, as well as the automatic peak marker and its harmonics.

To obtain meaningful results, the channel must be in vector trace mode. Due to slight errors when rescaling data after timebase zoom/translation, wait for one full left-to-right sweep before using the FFT results.

For best accuracy, set the timebase to collect a large (>10) number of oscillation periods, and use a high timebase resolution. The rectangular FFT window is preferred if the timebase spans an integer number of full periods.

Some measurements (such as THD) will display "Invalid" if there are not enough full periods to obtain an accurate measurement.

3.4.4   Data Export

The current trace for any channel can be exported using **File → Export**. This will open the export dialog shown below.
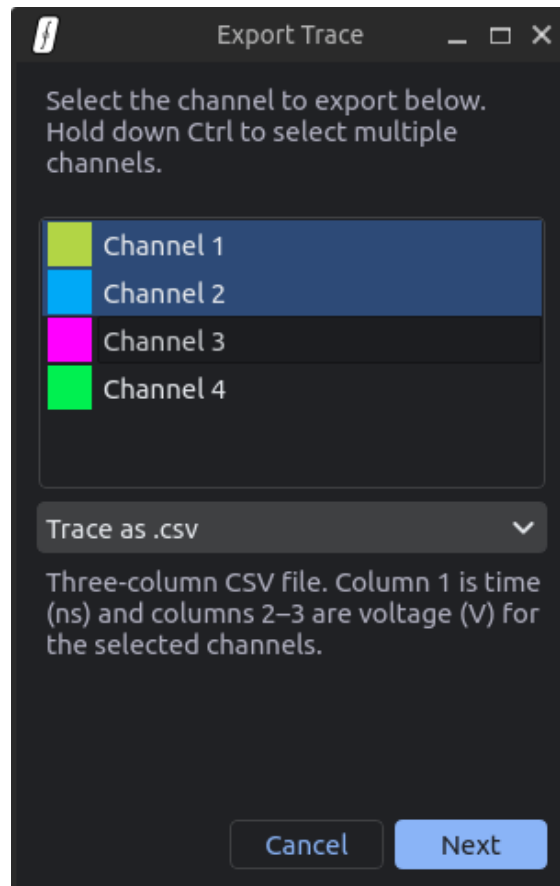
After choosing the channel(s) and format to export, a description of the export format will be shown. The save location can be chosen after clicking **Next**.

Note that acquisition will continue to run while the dialog is open, unless stopped beforehand.

Raw CDF Data: This export option provides access to the raw data returned from the scope. Data exported using this option can be later re-imported into the software, or analyzed externally.

The exported files are standard NumPy `.npz` archives, with keys:

n: The number of returned CDF entries at each time, for each channel ($N_\text{ch} \times N_\text{tb}$). For each index $0 \le \text{i} \le N_\text{tb}$, and channel $0 \le \text{j} \le N_\text{ch}$, use only the first first `ns[j,i]` entries of `vs[j,i,:]` and `ys[j,i,:]`.

vs: The voltages for the CDF samples at each time (up to 32), for each channel ($N_\text{ch} \times N_\text{tb} \times 32$).

ys: The CDF values for the CDF samples at each time (up to 32), for each channel ($N_{ch} \times N_{tb} \times 32$).

time: The timebase entries in nanoseconds (length N_{tb}).

Here N_{ch} is the number of selected scope channels for export, and N_{tb} is the number of points in the timebase (12 times the number of points per division).

3.4.5 Streaming to Disk

Data can be live-streamed to disk using **File ➜ Stream to Disk**. The "Stream to Disk" dialog shares the same options as the "Data Export" dialog.

After choosing a save filename (e.g. `filename.csv`), one file will be created at the end of each sweep in the chosen location. Each file will have a timestamp appended to the selected name (e.g. `filename-2024-01-21-18-55-04.csv`). Streaming can be stopped at any time by clicking "Stop" or closing the dialog.

3.4.6 Import and Overlay

Previously exported traces can be imported using **File ➜ Import**. Multiple traces can be imported from one file.

For single-valued traces, the selected file must be a `.csv` or `.npy` file whose first column is time (ns). Each additional column represents the corresponding voltages of one trace to import.

For multivalued signals (e.g. eye diagrams), the selected file must be a `.npz` file previously exported using the **Raw CDF Data** export option. If the selected file contains multiple signals, each one will be individually imported.

Each imported trace will appear as a newly created channel (labelled A, B, C,...). The "Timebase Shift" option in the extended settings can be used to shift the imported waveform to line up with events of interest.

3.5   Measurements

3.5.1   Basic Use

Measurements can be added by clicking the **+** icon at the bottom of the screen. A dialog will open allowing the desired measurement and channel to be selected.
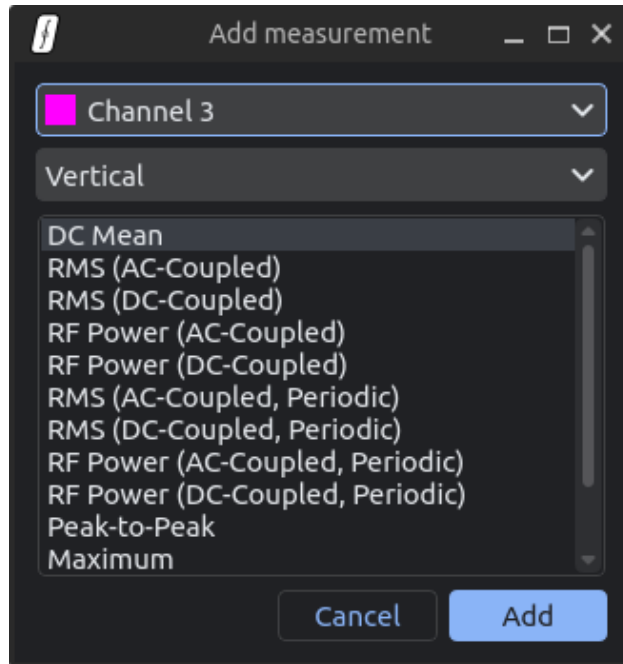
Tip: All periodic measurements (e.g. duty cycle, frequency) require vector trace to be enabled.

3.5.2   Active Measurement

One measurement for each channel is designated the **active measurement** and is highlighted. The active measurement can be changed by clicking on the desired measurement at the bottom of the screen.
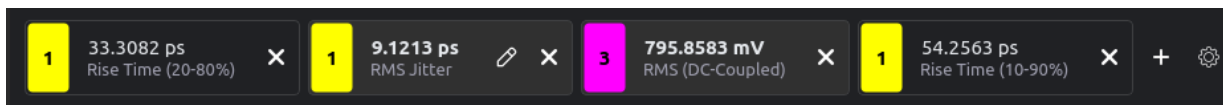
Adding a new measurement will automatically designate it the active measurement.

Associated information such as markers (e.g. risetime/falltime measurements), or the analyzed region and histogram (e.g. jitter measurements) are shown only for the active measurement.

3.5.3   Extended Measurement Information

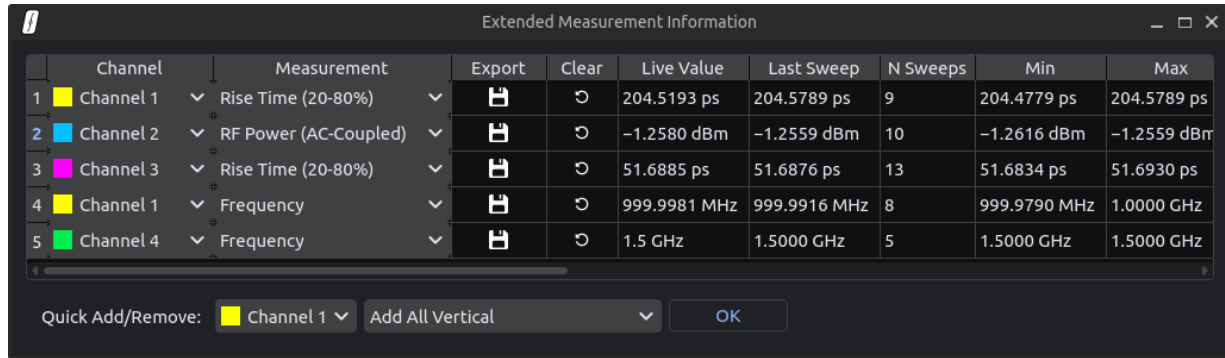Clicking the gear icon will open the **Extended Measurement Information** window. This allows all existing measurements to be edited, and tracks the statistics of each measurement over time.

All statistics update at the end of each full sweep. Note that changing the channel or type of a measurement will clear its statistics.

The currently tracked statistics are:

- Min (Minimum),
- Max (Maximum),
- Average,
- Stdev (Population Standard Deviation),
- Peak-to-Peak Deviation,
- Sample Stdev (Sample Standard Deviation),
- Median,
- Q1 (First Quartile),
- Q3 (First Quartile),
- IQR (Inter-Quartile Range).

The full sweep history for any statistic can be **exported** by clicking on the corresponding save icon. This will create a CSV file whose first column is a timestamp, and whose second column is the value.

The **Quick Add/Remove** tool can be used to add all basic vertical/horizontal measurements, or to delete all measurements, for a specified channel. Note that any existing measurements will be skipped (i.e. not duplicated) when using the quick add feature.

3.5.4   Jitter Measurements

Jitter measurements require selecting a region for analysis. When a jitter measurement is first created, a region is selected automatically based on the current timebase position and waveform range.

To manually define an analysis region, click on the **Edit** (pencil) icon. Right click to set one corner of the new analysis region, then right click again to define the opposite corner.

A histogram of the selected region will be displayed above the time axis.

For most accurate results, the height of the analysis region should be small relative to the slew rate of the signal. If the window is too tall, some smearing of the histogram will occur, and all jitter measurements will be overestimated.

When using the RMS Jitter or Rj($\delta\delta$) measurements, the contribution of trigger jitter should be subtracted using the formula

$$T_{\text{true}} = \sqrt{T_{\text{meas}}^2 - T_{\text{trig}}^2 - \alpha\Delta t},$$

where α and T_{trig} are provided on the calibration sheet. See Section 2.3 for details.

3.6   Math Channels

3.6.1   Overview

Add a math channel using **Tools → Add Math Channel**. After entering an arbitrary mathematical expression, press Enter to plot the function. All values are interpreted in volts.
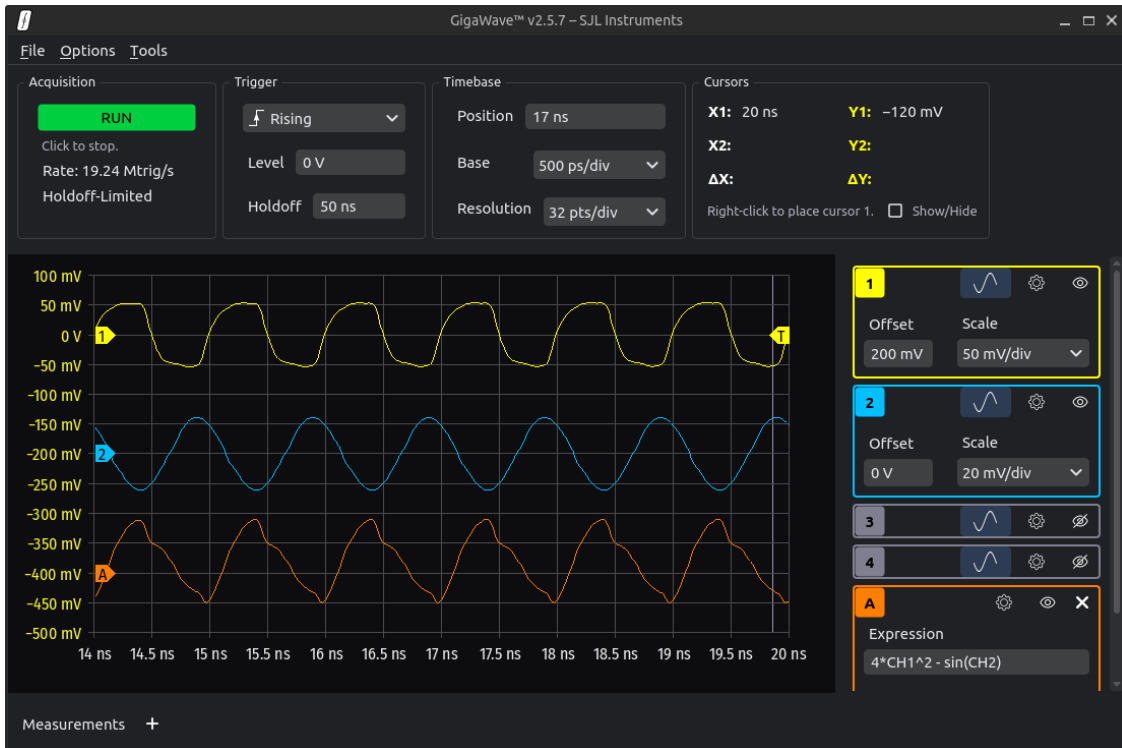
Reference any of the available channels using **CH1**, ..., **CH8**. The math feature can be combined with the de-embedding tool.

Currently supported functions are **sin**, **cos**, **tan**, **exp**, **ln**, **log10**, **sqrt**, and **abs**. Additional functions can be added upon request.

The variable **t** can be used to represent the time in nanoseconds. The constant **pi** (π) is defined for convenience.

Note: All referenced channels are assumed to be in **vector trace mode** (i.e. single-valued).

3.6.2   Min/Max Hold

In addition to the supported mathematical functions, there are special functions which keep track of the history over many sweeps. The currently supported functions are **maxhold**, **minhold**, and **average**. Additional functions can be added upon request.

For example, you can use the expression `maxhold(CH1)` to look for peaks in Channel 1 over many sweeps. At each post-trigger delay Δt, the largest observed Channel 1 voltage at that delay will be computed.

Note: The history of these functions will be cleared if the timebase position, zoom, or resolution is altered.

3.7 De-Embedding

3.7.1 Overview

The de-embedding utility can accessed by clicking on the **De-Embed** button in the extended settings for any channel. (The channel must be in vector trace mode.)

This tool can compensate for the distortion caused by any cables, splitters, or other sources of imperfection between the scope input and the DUT. In particular, de-embedding can counteract "dribble-up" when using a long coaxial cable as an analog delay line.

Due to the fundamental mathematics of this process, de-embedding is only possible in **vector trace mode** (i.e. the signal must be single-valued).

You can specify the system distortion either by:

(3.7.4) Importing measured S-parameters from a vector network analyzer (VNA), or

(3.7.5) Specifying the list of equipment between the DUT and the scope (e.g. the length and type of coaxial cables).

Measuring the S-parameters of your system will give the most accurate results.

The de-embedding settings are independent for each channel - different Touchstone files may be used for every channel, or a mix of the above two methods may be used.

Note: After changing timebase zoom/position/resolution, wait for one full left-to-right sweep before using the de-embedded results.

3.7.2 Example: Splitter and Analog Delay Line

To work around the 11 ns minimum trigger-to-capture delay of the GigaWave™, one may use the following setup:

- Signal from DUT sent into a resistive 6 dB splitter.

- Output 1 of the splitter sent directly into Channel 1 (trigger).

- Output 2 of the splitter sent through a long coaxial cable into Channel 2 (analog delay line).

The recommended de-embedding procedure is then as follows:

1. Connect Port 1 of a VNA to the input of the splitter. Connect Port 2 of the VNA to the far end of the coaxial cable. **Leave output 1 of the splitter connected** to Channel 1 – reflections from this portion will be compensated for by the de-embedding. Import the resulting Touchstone file into the Channel 2 de-embedding dialog.

2. Connect Port 1 of the VNA to the input of the splitter. Connect Port 2 of the VNA to output 1 of the splitter. **Leave output 2 of the splitter connected** to Channel 2 through the coaxial cable. Import the resulting Touchstone file into the Channel 1 de-embedding dialog.

The screenshot below shows example results from this procedure. We use a CCSMA18-MM-086F-120 coaxial cable and a ZFRSC-183-S+ resistive splitter, generating an analog delay of 14.4 ns.

- The orange trace (**A**) shows a 28 ps rising edge sent through the splitter and analog delay line, before de-embedding.

- The purple trace (**CH3**) is the live de-embedded result (note the factor of 2 in vertical scale).

- The red trace (**B**) is the rising edge seen through output 1 of the splitter (time-shifted to overlap with the other two traces).

Although not perfect, the de-embedded rising edge is significantly closer to the ground truth than the directly measured waveform.
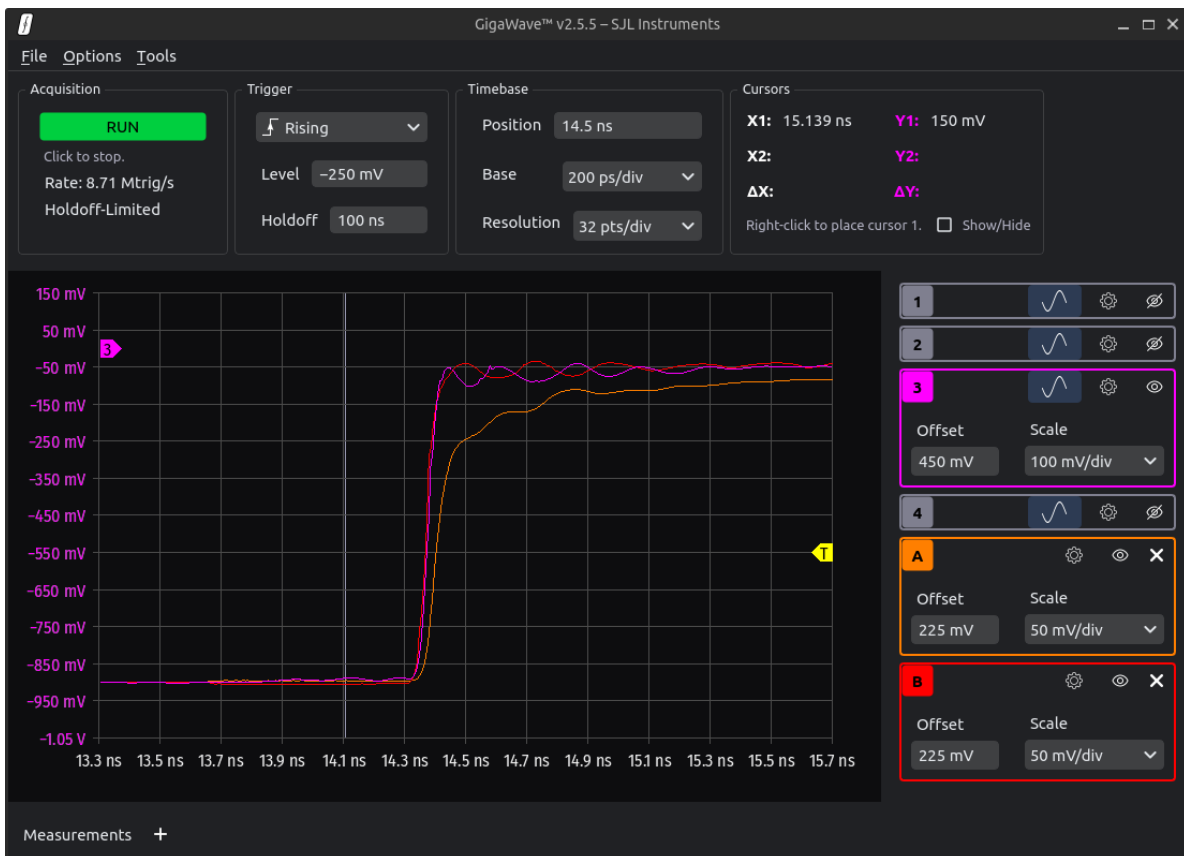
Note: Pre-measured Touchstone files for this example can be downloaded below, should you wish to recreate the setup without access to a VNA.

`https://gigawave-releases.s3.us-east-2.amazonaws.com/delayline-channel1.s2p`
`https://gigawave-releases.s3.us-east-2.amazonaws.com/delayline-channel2.s2p`

However, measuring your specific setup will give the most accurate results.

Larger-diameter coax cables will result in lower loss, and therefore less distortion. We have found LMR®-400 cables to give the best results while maintaining a cutoff above 6 GHz.

3.7.3   Capture Requirements

Accurate de-embedding of an event of interest (EOI) requires information from at least one rise-time **before and after** the EOI.
(If not known, the risetime is calculated automatically by the de-embedding tool.)

For example, a coaxial delay line may extend the risetime of a fast edge to 500 ps due to "dribble-up." Thus the timebase should be set to capture at least 500 ps on either side of the EOI for best results.

3.7.4   Importing S-parameters

To import S-parameters, the software requires a Touchstone (`.s2p`) file exported from a VNA. The software uses only the S21 parameter from this file.
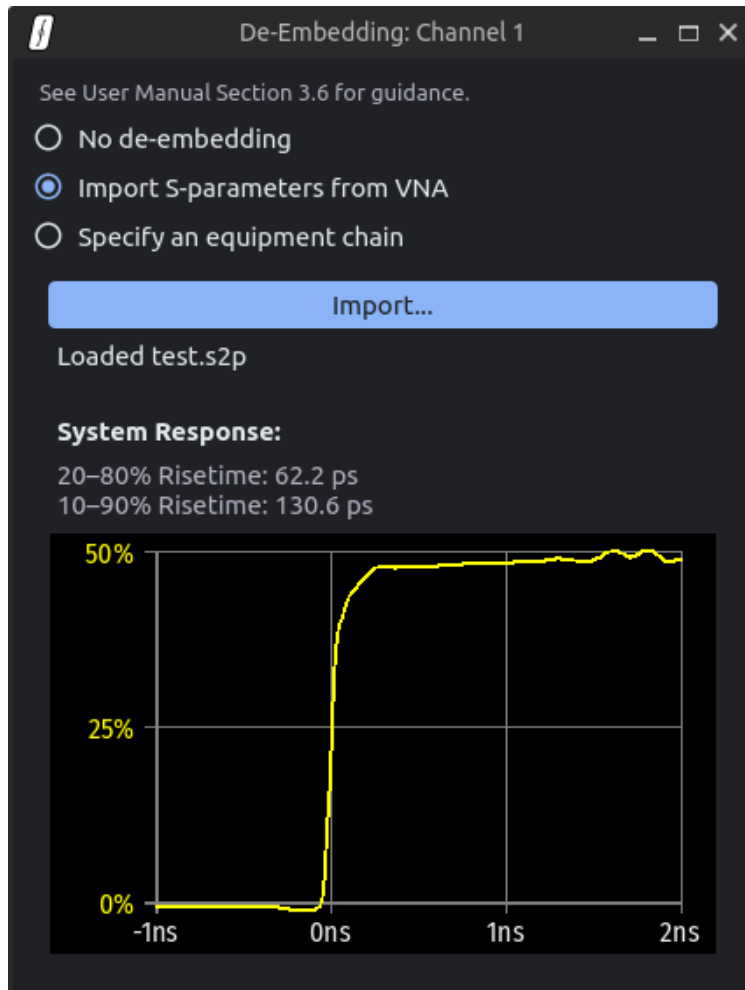
The VNA should be set up with Port 1 "in place of" the DUT, and Port 2 in place of the scope input. See Section 3.7.2 for an explicit example.

The frequency range should be set from **DC–6 GHz** (or higher) for best results. The upper end of

the VNA sweep will limit the risetime of the sharpest recoverable edge after de-embedding.

The lower end of the VNA sweep is not as critical – the software assumes that the S-parameters are constant between DC and the lowest provided frequency. This will lead to an error in the insertion loss (overall waveform scale) equal to the difference between the extrapolated and true S21 at DC. For most coaxial cables, this difference is small below 50 MHz.

3.7.5   Specifying an Equipment Chain

If a VNA is not available, you may also specify the list of equipment between the DUT and the scope input. This will give less accurate results than importing S-parameters.
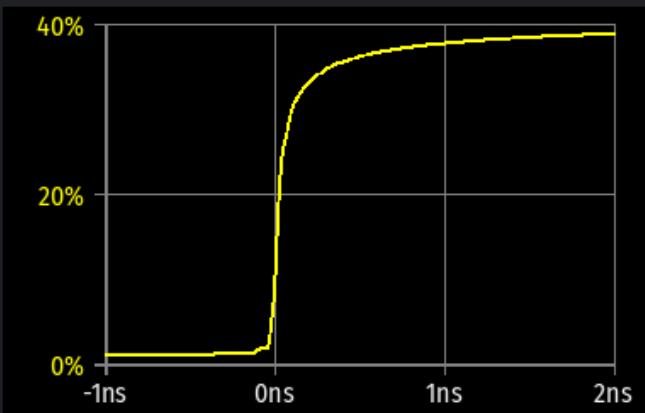
Each cable can be specified by type and length (in meters, feet, or inches), with a parameter for additional insertion loss. Each splitter or attenuator can be specified by its insertion loss (assumed flat across frequency).

The frequency response of each cable is computed according to the formula

$$\text{Attenuation (dB/100 ft)} = A_1\sqrt{f} + A_2 f$$

where A_1 (dB/100 ft/$\sqrt{\text{MHz}}$) and A_2 (dB/100 ft/MHz) are provided by the cable manufacturer. For RG-series cables, the parameters A_1 and A_2 follow the MIL-C-17 specification.

3.8   Other Tools

3.8.1   CDF Sampling Options

These settings can be accessed via **Options → CDF Sampling Options**. This dialog allows customization of the parameters $N_{\min}$, $N_{\max}$, and K defined in Section 2.2.1.

Detailed guidance on choosing these values is provided in Section 2.2.
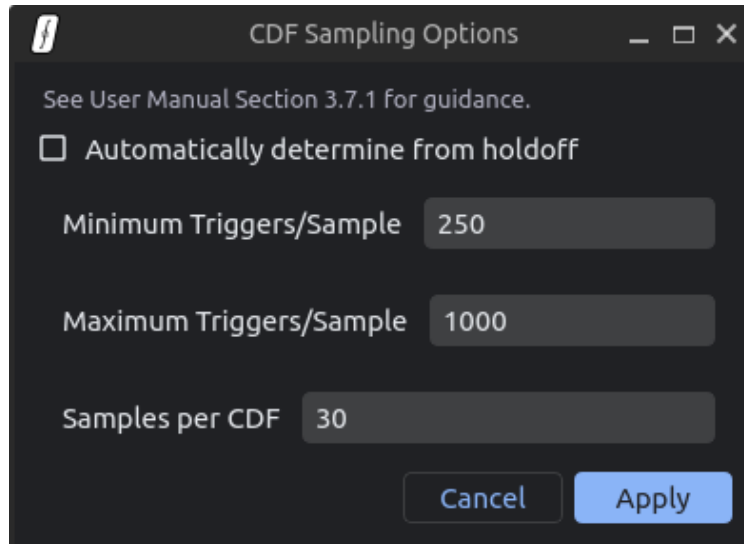
When the "Auto" box is checked, the values are determined automatically based on the holdoff, as described in Section 2.2.1.

3.8.2 Trim DC Offset

Note: If nothing is connected to the scope when the software is started, this procedure is executed automatically.

If a sub-mV DC offset is critical for your application, you can zero the offset via **Tools → Trim DC Offset**. After disconnecting all signals from the scope, click **OK** to remove the DC offset.

We recommend waiting until the scope has thermally settled before performing this procedure (5 minutes after power-on), as the offset may drift as a function of temperature.

The process will fail if the offset is larger than 10 mV. If the offset exceeds 10 mV, contact us for a diagnosis.

3.8.3 Device Information

The device serial number and firmware revision can be viewed using **Tools → Device Information**.
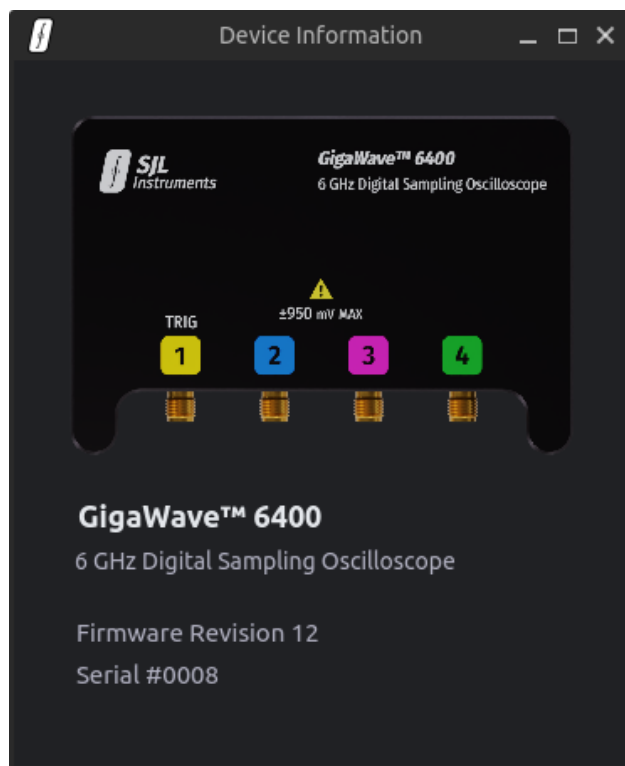
3.8.4   Automatic Setup

The vertical position and scale for all visible channels, as well as the trigger level, can be set automatically using **Tools → Auto Setup**. Although the timebase will not be set up automatically, the vertical autoscaling will function properly regardless of the timebase settings.

Selecting **Auto Setup (Overlay)** will overlay all channels to make best use of vertical space:
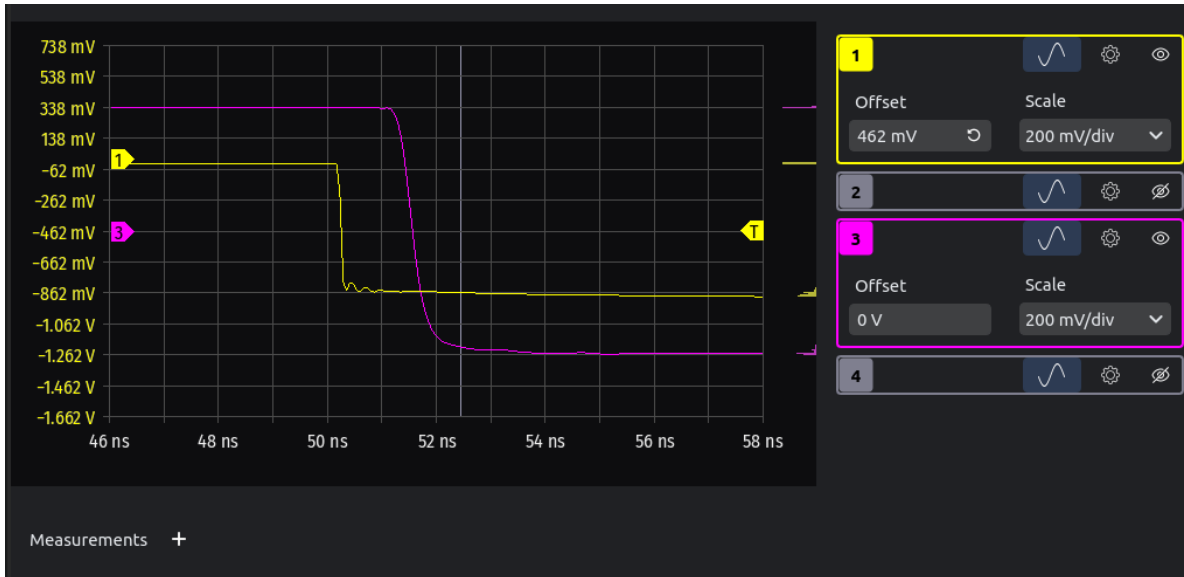
Selecting **Auto Setup (Stacked)** will stack all visible channels without vertical overlap:
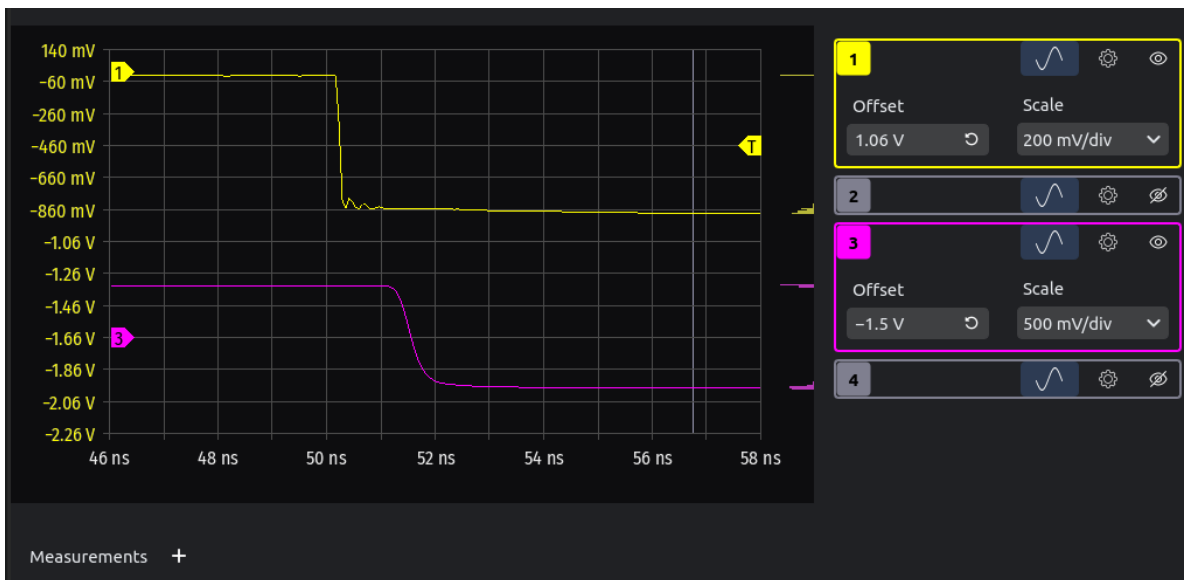
3.8.5 Power-User Mode

This mode is **intended for advanced users only**.

Power-user mode can be enabled via **Options → Power-User Mode**. This allows finer control over the timebase and CDF search range, as well as automatic calculation of the sampling timescale (Section 2.2.7).
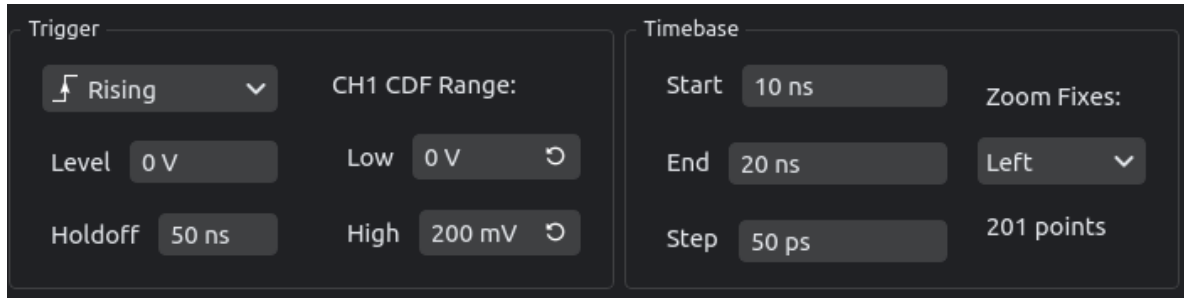
In this mode, the timebase start, stop, and step can be set manually. The timebase can be extended up to the smallest post-trigger delay allowed by the hardware; this is unit-dependent, but is generally 8 ns or earlier. Channel 1 will experience increased distortion in this region.

The CDF search range on Channel 1 can be adjusted to improve the trigger sensitivity. This can be helpful for triggering on small signals. For best sensitivity, set the range to be as as small as possible, while still containing the signal of interest (and the trigger level).

For small signals, the trigger level may need to be adjusted by a few millivolts to obtain a stable trigger.

Note: If the signal exceeds the CDF search range, the software will clip it to ±1.1 V.

3.8.6 Fast Mode

Note: Activating this option has no effect on firmware revisions 13 and lower.

Fast mode can be enabled via **Options → Fast Mode**.

When activated, the per-CDF overhead is reduced by ~8 ms at the expense of slightly degraded timing accuracy (~ 2 ps RMS). This mode should be activated only **after the device is in thermal steady-state** (5 minutes); activating it during warmup will lead to significantly degraded timing performance.

The speedup is most noticeable for single-valued signals (such as in a TDR setup), especially when low values of $N_{\min}$ or K are used. For TDR/TDT applications, setting $N_{\min} = 50$ and $K = 20$ leads to acceptable accuracy with sweep times below 2 seconds (16 points/division, one channel enabled).

3.8.7 Timebase Units (TDR/TDT)

The timebase can be displayed in equivalent distance units using **Options → Timebase Units**. This setting applies to the horizontal timebase labels and cursor readout.
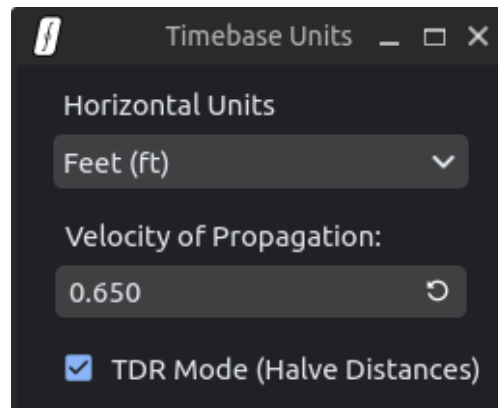
For TDR applications, enable the **TDR Mode** checkbox. This will halve all converted distances to correctly account for reflections.

4 Programming Guide

4.1 Overview

The GigaWave™ communicates via a standard USB serial interface at 921600 baud with 8 data bits, 1 stop bit, and no parity bits. All commands are **case-insensitive** and should be terminated with carriage-return only (`\r`). Responses will be CRLF-terminated (`\r\n`) unless otherwise noted.

Lowercase letters in the command format denote user-settable parameters. Uppercase letters should be sent exactly as shown.

To maintain the specified timing precision, the timebase must be calibrated at least once per second (Section 4.4.1). When using the serial interface, it is the user's responsibility to perform this calibration.

Any characters trailing the necessary parameters for a command will be ignored. The response `ERR RANGE` will be returned if the passed parameters fall outside the usable bounds, or an insufficient number of parameters are provided.[1]

After sending a command, you must wait for the corresponding response to begin before sending any subsequent command.[2] All commands will return a response. Malformed or unrecognized commands will return the response "`Unknown command!`".

---

[1]Firmware revisions 12 and below do not perform any range checks. Undefined behavior will occur if invalid parameters are used.

[2]For backwards compatibility purposes, commands sent while a previous command is processing will be queued into a FIFO. If many commands are queued, the device may appear unresponsive as the queue is processed sequentially.

4.2   Example Program

The following example program sets up a rising edge trigger at the level 0 V with 50 ns holdoff. It then measures the waveform on Channel 1 at 100 ps intervals between 15 ns and 20 ns.

```
E1                              (rising edge)
L32768                          (level 0 V)
H50                             (set 50 ns holdoff)
CAL                             (calibrate timebase)
D150000                         (set Δt = 15 ns)
R1 10000  10000  55000  55000   (acquire CDF data at Δt = 15 ns; set region of interest*)
D151000                         (set Δt = 15.1 ns)
R1                              (acquire CDF data at Δt = 15.1 ns)
  ⋮                               ⋮
D175000                         (set Δt = 17.5 ns)
R1                              (acquire CDF data at Δt = 17.5 ns)
CAL                             (calibrate timebase; ensure this occurs once per second)
D176000                         (set Δt = 17.6 ns)
R1                              (acquire CDF data at Δt = 17.6 ns)
  ⋮                               ⋮
D200000                         (set Δt = 20.0 ns)
R1                              (acquire CDF data at Δt = 20.0 ns)
```

The starred line (*) will need to be modified depending on your model number; the example shown is for a GigaWave™ 6200.

4.3   Interpreting CDF Data

4.3.1   Theory Review

The cumulative distribution function (CDF), denoted $F(V; \Delta t)$, gives the probability that the signal at the post-trigger delay Δt is less than or equal to the threshold V. For example, if measuring a sine wave, the function $F(V; \Delta t)$ at any fixed Δt will be a step function satisfying

- $F(V; \Delta t) = 0$, for $V < V_0$ and

- $F(V; \Delta t) = 1$, for $V \geq V_0$,

where V_0 is the actual voltage of the sine wave at the post-trigger delay Δt.

In practice, there will be a gradual transition from $F(V; \Delta t) = 0$ to $F(V; \Delta t) = 1$ depending on the noise present in the signal and the noise floor of the scope. As a first approximation, for single-valued signals (e.g. a simple periodic waveform), you can recover the signal voltage V_0 by interpolating between the two neighboring voltages where $F(V; \Delta t)$ first transitions from below $1/2$ to above $1/2$.

For multivalued signals (e.g. eye diagrams), you must take the derivative of $F(V; \Delta t)$ with respect to V to obtain the probability density function of the signal at Δt.

4.3.2   Example Analysis

As an example, the R command (read CDF) may return the following data:

Voltage (V)	CDF Value $F(V; \Delta t)$
−1.10000	0.000
0.00000	0.000
0.27500	0.000
0.41250	0.000
0.44688	0.000
0.46406	0.000
0.46836	0.000
0.47266	0.082
0.47480	0.352
0.47588	0.557
0.47695	0.749
0.47910	0.969
0.48125	0.996
0.55000	1.000
1.10000	1.000

For brevity, here we have set the samples per CDF K to 15 (see Section 2.2.1). By eye, we can see that the corresponding voltage for this post-trigger delay Δt is around 476 mV, the location where

the CDF reaches 0.5. The noise in the signal is on the order of 1 mV, the voltage scale over which the CDF varies from 0.3 to 0.7.

If the signal is single-valued, a simple way to extract the corresponding voltage is to linearly interpolate between the two voltages `0.47480 V` and `0.47588 V`, the two neighboring entries where the CDF crosses 0.5. This would give

$$(0.5 - 0.352) \cdot \frac{0.47588 \text{ V} - 0.47480 \text{ V}}{0.557 - 0.352} + 0.47480 \text{ V} = \boxed{0.47558 \text{ V.}}$$

Note that the specified ENOB may not be achieved by this technique due to its poor noise rejection. The desktop software uses a global nonlinear fit that reduces the noise sensitivity (Section 4.3.5).

4.3.3 CDF Monotonicity Correction

Note: Firmware revisions 14 and later perform this correction automatically.

Due to inherent statistical noise in the sampling process (see Section 2.2.5), the returned CDF values may not be increasing as a function of voltage ("monotonic"). The level of noise may be significantly improved by enforcing monotonicity on the CDF.

A simple algorithm to perform this correction is as follows:

1. Search for any adjacent values that are out of order; i.e. two voltages $V_1 < V_2$ with corresponding CDF values $F(V_1) > F(V_2)$.

2. Compute the average $F_{\text{avg}} = (F(V_1) + F(V_2))/2$ and deviation $\Delta F = F(V_1) - F(V_2)$.

3. Overwrite the value $F(V_1)$ with $F_{\text{avg}} - \eta \Delta F$, and overwrite $F(V_2)$ with $F_{\text{avg}} + \eta \Delta F$.

4. Repeat the above steps until no pairs are out of order.

Here the "overcorrection parameter" η lies between 0 and 0.5. A lower value of η will be less invasive, but will require more iterations to converge. The official software uses $\eta = 0.1$.

4.3.4   Interpolation Pitfalls

When using the linear interpolation method with low K (samples per CDF), the extracted voltage may experience occasional errors.

An exaggerated example is shown below. Although the returned CDF values are correct, the interpolation method mis-estimates the mean voltage.
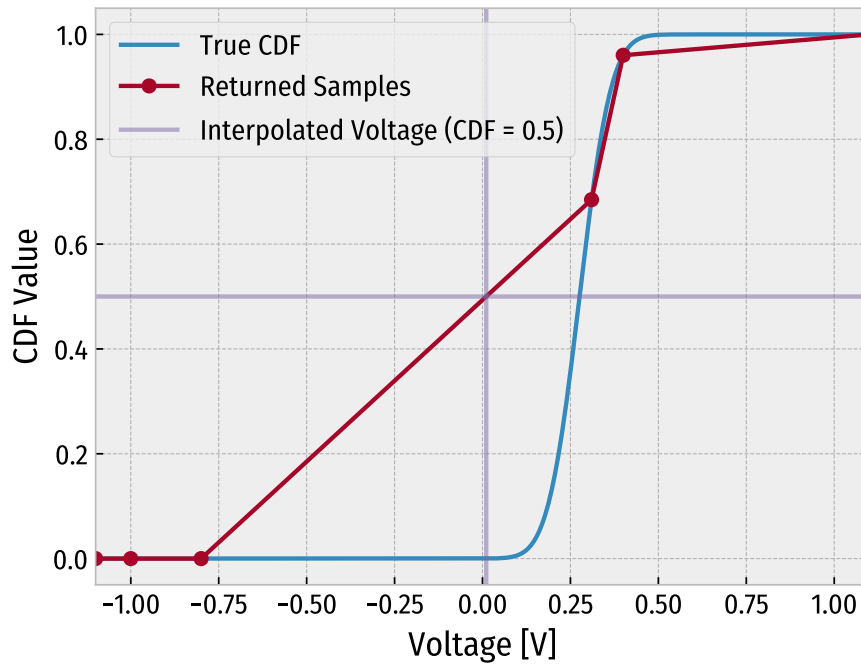
This effect can be avoided by keeping $K > 20$, or by instead fitting a Gaussian error function to extract the mean voltage (Section 4.3.5).

4.3.5 Error Function Fit

For most accurate results, and to avoid the effect shown in Section 4.3.4, the software fits a Gaussian error function to the CDF to extract the central voltage. This is the theoretical CDF of a unimodal probability distribution with normally-distributed noise.

The fit form is given by:

$$f(x) = \frac{1}{2}\left[1 + \mathsf{erf}\left(\frac{x - x_0}{\sigma\sqrt{2}}\right)\right],$$

where x_0 and σ are free parameters giving the central voltage and noise scale, respectively. Here, the standard Gaussian error function **erf** is defined as

$$\mathsf{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\,dt,$$

and is available as a built-in function in most analysis software.

The plot below shows this fit applied to the example data from Section 4.3.2. The extracted mean values are $x_0 = 475.573$ mV and $\sigma = 2.037$ mV.
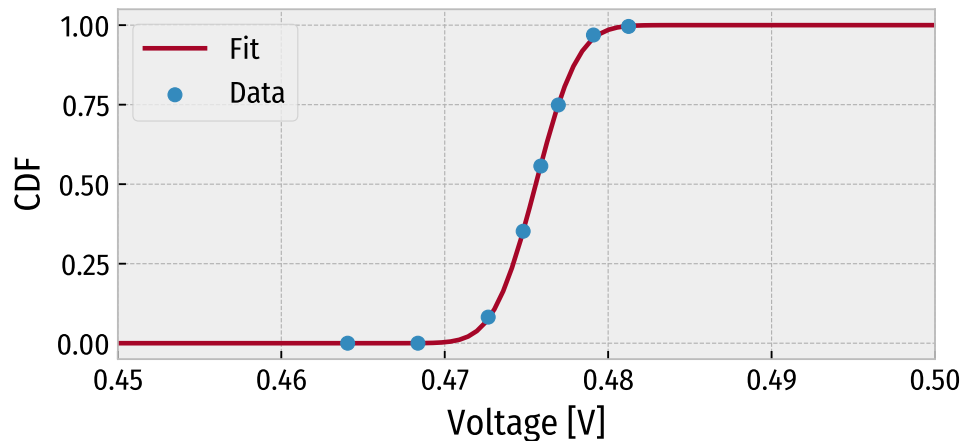
Implementation Note 1: The official software linearizes the above fit for computational efficiency. Rather than fitting $f(x)$ directly to V, F data, it performs a weighted fit of a line to $V, g(F)$ data, where $g(x) = \mathsf{erf}^{-1}(2x - 1)$ is the inverse Gaussian error CDF. The values of $g(F)$ **must** have their weights scaled by $(g'(F))^{-2}$ for proper error propagation and fit stability. The fit is performed only on the subset of entries satisfying $0.1 < F < 0.9$ for numerical stability.

Implementation Note 2: If performing the fit directly (without linearization), the fit will likely fail unless good seed parameters are provided. We recommend using linear interpolation to seed x_0, and choosing 10 mV as an initial σ.

4.4   Core Commands

4.4.1   CAL (Calibrate Timebase)

Command Format: `C` or `CAL`
Expected Response: `OK CAL`
Available in all firmware revisions.

This command calibrates the timebase. Calibration usually takes 60 ms.

The 1 ps timing precision of the GigaWave is guaranteed only up to one second after this command is issued. We therefore recommend issuing this command at least once per second. If data acquisition (`R`) takes more than one second, the `CAL` command must be issued immediately before the corresponding delay (`D`) and data acquisition (`R`) commands.

If the device is in thermal steady-state ($\sim$5 minutes after warmup), the 1 ps precision can often be maintained for up to 30 seconds without recalibration. If calibrating at a reduced frequency, the 1 ps specified accuracy is not guaranteed, and it is the user's responsibility to verify that the timing precision meets the application requirements.

Note: On firmware revision 12 and below, performing two back-to-back calibrations (issuing `CAL` twice in a row) may lead to an error when the next delay (`D`) command is issued.

4.4.2   D (Set Post-Trigger Delay)

Command Format: `Dx y`
Expected Response: `OK D`
Available in all firmware revisions.

This command sets the post-trigger delay Δt at which sampling will occur. The delay is specified as an integer, with one unit representing 0.1 ps.

The minimum achievable delay is unit-dependent, but is generally 8 ns (`D80000`) or lower. Note that Channel 1 will experience increased distortion in the 8 to 11 ns region.

If a `CAL` command is issued, any previously specified delay is **invalidated** and this command must be re-sent.

Sending a `D` command will also reset the trigger source to external (equivalent to an `X0` command), regardless of whether the command succeeds.

Internally, the delay will be measured and fine-tuned to achieve an accuracy of 0.2 ps or better. This process takes $\sim$8 ms. If the optional argument `y` is set to 1, the verification will be skipped, saving time in exchange for a typical accuracy of $\sim$2 ps.

Note: The parameter `y` is ignored in firmware revisions v13 and earlier.

Error Response: `WARNING: Delay underflow -inf`
This error occurs when the requested value cannot be achieved by the delay generator. If you are certain the requested delay falls within acceptable bounds, contact us for a diagnosis.

4.4.3   R (Acquire CDF)

Command Format (6200): `Rx s0 s1 e0 e1 y`
Command Format (6400): `Rx s0 s1 s2 s3 e0 e1 e2 e3 y`
Command Format (6800): `Rx s0 s1 s2 s3 s4 s5 s6 s7 e0 e1 e2 e3 e4 e5 e6 e7 y`
Expected Response: `OK a b c d xSJLI.`
Available in all firmware revisions.

This command returns CDF data in a binary format. You must specify the desired post-trigger delay Δt and set up the trigger appropriately before issuing this command.

Parameters:

- The parameter `x` is a bitmask for the channels to be returned. For instance, the command `R19` would return data from only Channels 1, 2, and 5 (since 19 = 0b00010011).

- The parameters `s0`…`s7` represent the lower bound ("start voltage") of the region to sample for each channel, on a scale from –1.5 V (`0`) to +1.5 V (`65535`). This will be the lowest voltage returned in the corresponding CDF data. The number of parameters required depends on the model of the oscilloscope, and is equal to the number of channels (regardless of the bitmask).

- Likewise, the parameters `e0`…`e7` represent the upper bound ("end voltage") of the region to sample on each channel, and will be the highest voltage returned in the corresponding CDF data.

- The parameters `s0`…`s7` and `e0`…`e7` are optional on subsequent invocations, and will retain previously set values if omitted. Modifying them can be useful for obtaining higher vertical resolution when the samples per CDF K is low.

- [Firmware v14 and later] The optional parameter `y` determines the CDF return format. If `y` is set to 2, then the returned CDF values have 16-bit precision. Otherwise, 8-bit precision is used.

The search region (`e0 - s0`, `e1 - s1`, etc.) for any channel should not be smaller than 200 units ($\sim$9 mV).

Note that decreasing the search region for Channel 1 (`e0 - s0`) will improve the sensitivity of the trigger to low-amplitude signals, if the region contains the trigger level.

Response:
The response is **not CRLF-terminated**.

The response contains four integers (`a`, `b`, `c`, `d`) followed by binary data (`x`). The four integers are:

- The total number of triggers acquired.

- The total time (in microseconds) spent acquiring data.

- The total wall-clock time spent (in clock cycles @ 96 MHz) including communication time and other overhead.

- The number of CDF samples (per channel) contained in the binary data.

The trigger rate (in Mtrig/s) can be determined by taking the ratio `a`/`b`.

If `y` is omitted, the binary data consists of `d` chunks of $3N_{ch}$ bytes each, where N_{ch} is the number of channels requested via the bitmask. Each chunk contains N_{ch} three-byte entries, each representing a single CDF value $F(V; \Delta t)$ for each requested channel.

The first two bytes of each entry represent the voltage V on a scale from –1.5 V (`0x0000`) to +1.5 V (`0xFFFF`) in big-endian format. The third byte represents the value $1-F(V; \Delta t)$, multiplied by 255 (i.e. a returned value of `0` corresponds to $F(V; \Delta t) = 1$, and `255` corresponds to $F(V; \Delta t) = 0$).

If `y` is set to 2, then each chunk contains $4N_{ch}$ bytes, representing N_{ch} four-byte entries. The first two bytes in each entry represent the voltage V as before. The last two bytes represent the value $1 - F(V; \Delta t)$ multiplied by 65535, in big-endian format.

Example:
If the command `R9` was sent, and the response sets `d` = 3, then the returned binary data should be interpreted in the following order:

$$\underbrace{V_{1,1} \quad F_{1,1} \quad V_{1,4} \quad F_{1,4}}_{\text{Chunk 1}} \quad \underbrace{V_{2,1} \quad F_{2,1} \quad V_{2,4} \quad F_{2,4}}_{\text{Chunk 2}} \quad \underbrace{V_{3,1} \quad F_{3,1} \quad V_{3,4} \quad F_{3,4}}_{\text{Chunk 3}},$$

where $V_{n,c}$ (2 bytes) is the nth voltage entry for channel c, and $F_{n,c}$ (1 byte) is the corresponding CDF entry. The CDF data for Channel 4 would then consist of the data $(V_{1,4}, F_{1,4}), (V_{2,4}, F_{2,4}), (V_{3,4}, F_{3,4})$. See Section 4.3.2 for an example of how to process this data.

Error Response: `NO TRIG ZERO SJLI`
Error Response: `NO TRIG LOW x SJLI`
These errors occur when no or too few triggers occurred for accurate CDF measurement. Please verify that the trigger source is present and the trigger settings are correctly initialized.

These errors may also occur when the requested post-trigger delay is outside the acceptable range defined by the trigger holdoff.

Error Response: `NO TRIG TIMEOUT SJLI`
This error occurs when the time taken for CDF sampling exceeds the maximum timeout interval. This can occur when N_{max} is too high with a low-rep-rate trigger source (see Sections 2.2.1 and 3.8.1 for guidance).

4.4.4   @ (Direct CDF Read)

Command Format (6200): `@v0 v1`
Command Format (6400): `@v0 v1 v2 v3`
Command Format (6800): `@v0 v1 v2 v3 v4 v5 v6 v7`
Example Response (6400): `OK a0 b0 a1 b1 a2 b2 a3 b3 SJLI`
Available in all firmware revisions.

Directly probes the values $F(V; \Delta t)$ for each channel. The parameters `v0`…`v7` are the query voltages V for each channel, on a scale from –1.5 V (`0`) to +1.5 V (`65535`).

You must configure the appropriate post-trigger delay Δt, trigger settings, and CDF sampling settings before issuing this command.

The returned integers `a0`…`a7` represent the number of triggers where the signal exceeded the query voltage V, on each channel. Similarly, the returned integers `b0`…`b7` represent the number of triggers where the signal did not exceed the query voltage V.

The CDF value $F(V; \Delta t)$ for Channel 1 can be calculated as `b0`/(`a0` + `b0`), and likewise for the other channels.

The response is **not CRLF-terminated**.

Error Response: The possible error responses are the same as those for the read CDF (`R`) command (Section 4.4.3).

Note: Use this command only when necessary, as the read CDF (`R`) command will have lower overall communication overhead.

Note: The sums (`a0` + `b0`), …, (`a7` + `b7`) may differ by up to four. This does not bias the CDF results.

4.5 Trigger Commands

4.5.1 H (Set Trigger Holdoff)

Command Format: `Hx`
Example Response: `OK H100ns.`
Available in all firmware revisions.

Sets the trigger holdoff in nanoseconds. The minimum value is 40. The maximum value is 21000.

4.5.2 L (Set Trigger Level)

Command Format: `Lx`
Expected Response: `OK Lx.`
Available in all firmware revisions.

Sets the trigger level on a scale from –1.5 V (`L0`) to +1.5 V (`L65535`).

4.5.3 E (Set Trigger Direction)

Command Format: `Ex`
Expected Response: `OK Ex.`
Available in all firmware revisions.

Sets the trigger direction to rising (`E1`) or falling (`E0`).

4.5.4 X (Set Trigger Source)

Command Format: `Xx`
Expected Response: `OK Xx.`
Available in all firmware revisions.

Sets the trigger source to internal (`X1`) or external (`X0`). The default source is external. The internal trigger source (>100 Mtrig/s) is not synchronized to any external signal.

Issuing a `D` (delay) command will also reset the trigger source to external.

When the internal trigger source is selected, all other trigger settings (level, direction, holdoff) are not used.

4.6   CDF Sampling Settings

See Section 2.2.1 for an explanation of these parameters, and Section 3.8.1 for guidance on choosing appropriate values.

4.6.1   S (Set Minimum Triggers per Sample)

Command Format: Sx
Expected Response: OK Sx.
Available in firmware revision 10 and above.

Sets the minimum triggers per sample N_{min}.

The default value is 4096. The minimum value is 3. The maximum value is 4 billion.

4.6.2 $ (Set Maximum Triggers per Sample)

Command Format: $x
Expected Response: OK $x.
Available in firmware revision 11 and above.

Sets the maximum triggers per sample N_{max}.

The default value is 65535. The minimum value is 3. The maximum value is 4 billion.

4.6.3   Q (Set Samples per CDF)

Command Format: Qx
Expected Response: OK Qx.
Available in firmware revision 11 and above.

Sets the number of samples per CDF K.

The default value is 30. The minimum value is 3. The maximum value is 250.

4.6.4 % (Set CDF Tolerance)

Command Format: %x
Expected Response: OK %x.
Available in all firmware revisions.

Sets the CDF sampling tolerance ϵ, in ppm. The default value is $\epsilon = 0.01$ (corresponding to %10000).

Sampling will stop if **either**:

- The statistical uncertainty on the CDF $F(V; \Delta t)$ falls below ϵ, or

- The maximum triggers per sample $N_{\max}$ is reached.

See Section 2.2.1 for more details.

Note: In firmware revision 14 and below, the timeout of the read (R) command is scaled inversely to the CDF tolerance. Setting a sampling tolerance below 100 ppm may lead to an apparent hangup as the device attempts to collect a large number of triggers. This behavior will be fixed in firmware revision 15.

4.7 Auxiliary Commands

4.7.1 ^ (Read Device Info)

Command Format: ^
Example Response: `11 8 0.0000 0.0000 0.0000 0.0000 1.0000 1.0000 1.0000 1.0000`
Available in all firmware revisions.

Returns the device firmware revision and serial number (revision 11 and SN #0008 in the example response). Other values are reserved.

4.7.2 *IDN? (Get Device Identifier)

Command Format: `*IDN?`
Example Response: `GigaWave 6400 Digital Sampling Oscilloscope`
Available in all firmware revisions.

Returns a human-readable device name.

4.7.3 *TST? (Self-Test)

Command Format: `*TST?` or `T`
Expected Response: `POST passed`
Available in all firmware revisions.

Performs a self-test of all channels.

If successful, all channels generate correct data using the internal trigger source, and the time-base generator functions correctly. The self-test **does not** verify the functionality of the external trigger path.

Possible error responses and diagnoses follow. Please contact us if you require assistance with any of the below instructions.

FPGA communication failure: The MCU cannot establish a connection to the FPGA. The FPGA firmware may be corrupt. Reinstall the FPGA firmware using the "legacy" instructions in the firmware update guide.

BAD MAX/MAX DELAY: Measured range of the delay generator does not meet the nominal specifications. Please contact us for assistance, as there are multiple possible causes.

LOW/HIGH SAMPLE FAILED: Data was not received from the front-end hardware. Please contact us for assistance, as there are multiple possible causes.

CHx BAD: The specified channel is always railed high or low. This can occur due to (a) a connected signal which exceeds the absolute maximum rating, (b) an ESD-damaged front-end chip, or (c) malfunctioning query voltage generator. In case (a), the problem can be resolved by disconnecting the signal. In cases (b) and (c), contact us for further instructions.

4.8   Device Firmware Update

The device firmware update (DFU) serial protocol is documented for reference purposes only. **We recommend using the official software to perform firmware updates.**

Flashing malformed custom firmware via this protocol may put the device into an unusable state. If this occurs, the device will need to be recovered using the "Legacy" update procedure documented in the Firmware Upgrade Guide.

4.8.1   MCU Firmware Update

Immediately after power-on, the device will enter a bootloader which listens for DFU packets. If no valid packet is received within two seconds, the bootloader will proceed to run the main firmware.

A valid MCU DFU packet consists of 262 bytes, arranged in the following format:

- The first byte is the **message type** and must be one of the characters E, W, or J.

- The next four bytes represent a 32-bit **address** in flash memory space, in big-endian format.

- Bytes 6–261 represent a 256-byte binary **payload**.

- The last byte is a **parity checksum**, equal to the bitwise XOR of bytes 1–261.

After receiving a string of 262 bytes, the bootloader will respond with 0xBB (ACK) if the packet is valid and was successfully processed, or 0x6A (NAK) otherwise.

The message type is parsed as follows:

- Message type E erases all flash sectors containing the main MCU firmware. The address and payload are ignored.

- Message type W writes the provided 256-byte payload to the specified address.

- Message type J jumps to the reset handler in the main MCU firmware (0x80004004). The address and payload are ignored. This command should be issued once all writes are complete.

The bootloader occupies memory locations 0x80000000 through 0x80003FFF and should not be overwritten. The main firmware should be written starting from address 0x80004000.

4.8.2   FPGA Firmware Update

Update of the FPGA firmware is controlled by the MCU. The MCU firmware therefore defines the serial protocol for this process.

We currently do not document the details of this protocol, as we plan to substantially change the FPGA DFU packet structure in the next MCU firmware revision (v15). This section will be updated with packet-level documentation after v15 is released.