HydraHarp 500

PicoQuant

High Resolution
Multichannel TCSPC Systems and
Time Taggers with USB Interface

User's Manual and Technical Data

Version 1.0.0.0

Disclaimer

License and Copyright Notice

Acknowledgments

The HydraHarp 500 hardware in its current version as of August 2025 uses the White Rabbit PTP core v. 4.0 licensed under the CERN Open Hardware License v1.1 and its embedded WRPC software licensed under GPL Version 2, June 1991. For Details please see https://ohwr.org/projects/white-rabbit/ and the links to license terms and related documents there. The WRPC software used in the HydraHarp 500 was minimally modified and in order to meet the licensing terms the modified WRPC source code is provided as part of the HydraHarp 500 software distribution download.

When the HydraHarp 500 software is used under Linux it uses Libusb to access the HydraHarp 500 USB devices. Libusb is licensed under the LGPL which allows a fairly free use even in commercial projects. For details and precise terms please see http://libusb.info. In order to meet the license requirements a copy of the LGPL as applicable to Libusb is provided as part of the HydraHarp 500 software distribution pack. The LGPL does not apply to the HydraHarp 500 software as a whole.

Table of Contents

2. Introduction

While intensity based fluorescence spectroscopic investigations have been fairly common for a long time, extracting additional temporal information from quantum systems via pulsed excitation and time-resolved detection is a relatively new and powerful technique. The temporal analysis can reveal information about the emitter that is not available from spectral data alone. This is why time-resolved analysis of (typically laser induced) fluorescence by means of Time–Correlated Single Photon Counting (TCSPC) has gained in importance over the recent years. For instance, in life sciences the difference in the fluorescence decay times of fluorophores provides a powerful discrimination feature to distinguish molecules of interest from background or other species. This has made the technique very interesting for sensitive analysis, even down to the single molecule level. The same mechanisms are applicable in quantum optics, e.g., when quantum dots or defect centers in diamond are observed.

The acquisition of fluorescence decay curves by means of TCSPC provides resolution and sensitivity that cannot be achieved with other methods. In practice it is done by histogramming arrival times of individual photons over many excitation and fluorescence cycles. The arrival times recorded in the histogram are relative times between excitation and corresponding fluorescence photon arrival (start / stop times) ideally resolved down to a few picoseconds. The resulting histogram represents the fluorescence decay. Although fluorescence lifetime analysis is a great field of application for the HydraHarp 500, it is in no way restricted to this task. Other important applications are e.g. quantum optics, Quantum Cryptography (QC) Time–Of–Flight (TOF) and Optical Time Domain Reflectometry (OTDR) as well as any kind of coincidence correlation.

The HydraHarp 500 is a cutting edge TCSPC and time tagging system with USB interface. Its integrated design provides a flexible number of input channels at reasonable cost and allows innovative measurement approaches. The timing circuits of each channel allow high measurement rates up to 85 million counts per second (Mcps) with an excellent time resolution and a record breaking dead time of only 680 ps. The USB 3.0[1] super speed interface provides very high throughput as well as 'plug and play' installation. The input triggers are adjustable for a wide range of input signals providing programmable edge+level triggers for both negative and positive going signals and model dependent also Constant Fraction Discriminators (CFD) for negative going signals. These specifications qualify the HydraHarp 500 for use with all single photon detectors, notably Single Photon Avalanche Diodes (SPADs), Photomultiplier Tubes (PMT), and superconducting nanowire detectors.

The HydraHarp 500 can be purchased in different versions with currently up to 17 timing inputs of which one is suitable as a synchronization (sync) input for very high frequencies. The use of the inputs is very flexible. In fluorescence lifetime applications the sync channel is typically used as a timing reference input from a laser. The other inputs are then used for photon detectors. In coincidence correlation applications all inputs including the sync input can be used for photon detectors.

The HydraHarp 500 can operate in various modes to adapt to different measurement needs. The standard histogram mode performs real–time histogramming in on-board memory. Two different Time–Tagged–Time–Resolved (TTTR) modes allow recording of each photon event on separate, independent channels, thereby providing unlimited flexibility in off–line data analysis such as burst detection and time–gated or lifetime weighted Fluorescence Correlation Spectroscopy (FCS) as well as picosecond coincidence correlation, using the individual photon arrival times. The HydraHarp 500 is furthermore supported by a variety of accessories such as pre–amplifiers, signal adapters and detector assemblies from PicoQuant. A significant novel feature of the HydraHarp 500 is support for White Rabbit, allowing time transfer and synchronization with sub-ns accuracy (see https://en.wikipedia.org/wiki/The_White_Rabbit_Project).

The HydraHarp 500 software runs on current x86-64 Windows PC platforms. It provides functions for setting measurement parameters, displaying measurement results, loading and saving of measurement parameters and decay curves as well as the recording of time tagged photon event data. Important measurement characteristics such as count rate, count maximum and position, histogram width (FWHM) are displayed continuously. Data can conveniently be exported via the clipboard, e.g. for immediate processing by the EasyTau 2 Fluorescence Decay Fit Software. Furthermore, a programming library (DLL) enables users to write custom data acquisition programs for the HydraHarp 500 in virtually all popular programming environments. There is also a library version for Linux (AMD/Intel x64 processor architecture only) which is fully compatible with that for Windows so that applications can easily be ported across the two platforms.

Another relatively advanced high-level API package "snAPI" for Python is also available (Windows only). It readily provides data collection and file writing methods as well as many real-time analysis methods such as intensity and coincidence time traces, FCS and $g^{(2)}$ correlation.

1 The latest USB specifications have introduced new naming schemes where the original USB 3.0 (5 GBits/s) was later called "USB 3.1 Gen 1" and now "USB 3.2. Gen 1x1". We stick to the original name USB 3.0 throughout this document.

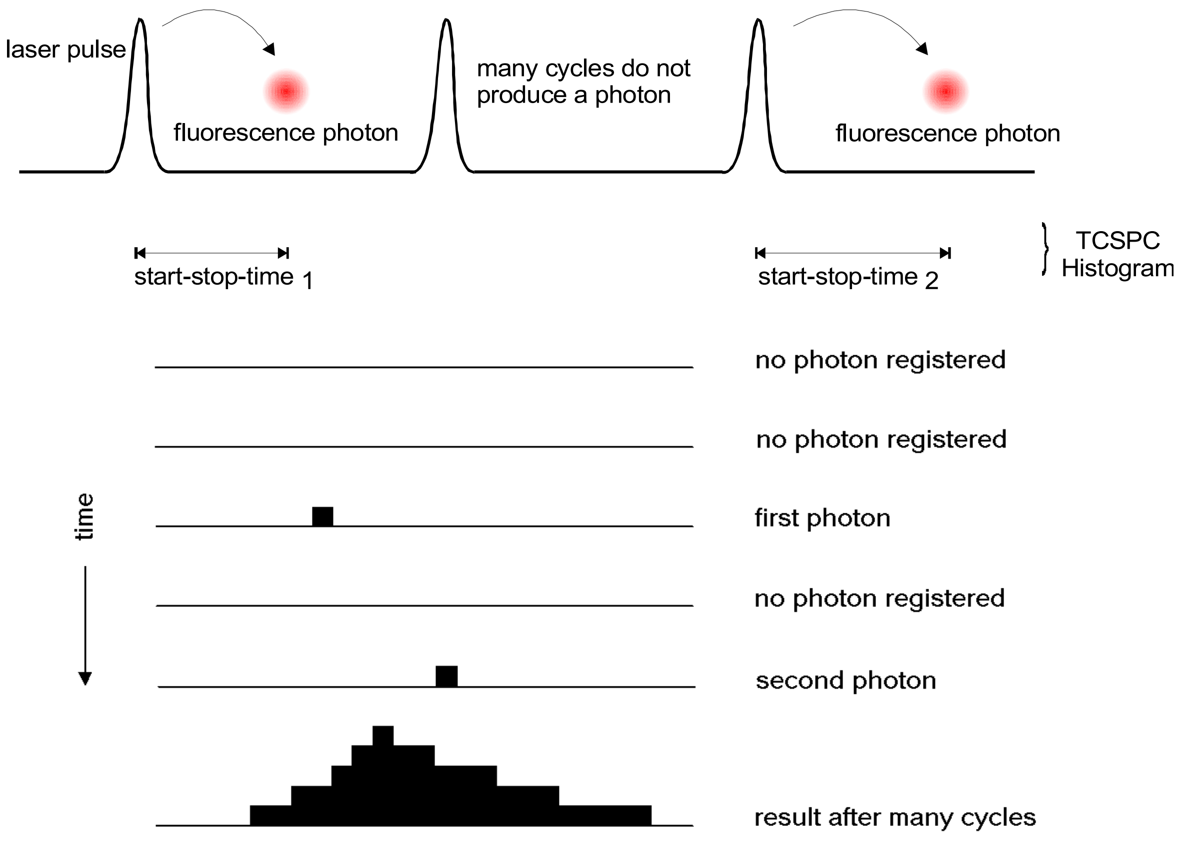
Yet another software suite that can be used with the HydraHarp 500 is the relatively new UniHarp software available for download from the PicoQuant web site. It provides convenient data collection and file writing as well as subset of the advanced real-time analysis methods available in the snAPI suite mentioned above.

For details on the Time–Correlated Single Photon Counting method, please read the next section as well as our TechNote on TCSPC and consult the literature referenced at the end of section 3.4. Experienced users of the method should be able to work with the HydraHarp 500 straight away. Nevertheless, we recommend carefully reading sections 4.4 and 4.5 on hardware and software installation to avoid damage. Later, the comprehensive online–help function of the HydraHarp 500 software will probably let the manual gather dust on the shelf.

3. Primer on Time–Correlated Single Photon Counting

In order to make use of a powerful analysis tool such as time–resolved fluorescence spectroscopy, one must record the time dependent intensity of the emitted light. While in principle, one could attempt to record the intensity decay of the signal from a single excitation / emission cycle, there are practical problems that prevent such a simple solution in most cases. First of all, the decay to be recorded is very fast. Typical fluorescence from organic fluorophores lasts only a few hundred picoseconds to some hundred nanoseconds. In order to recover fluorescence lifetimes as short as e.g., 500 ps, one must be able to resolve the recorded signal at least to such an extent, that the exponential decay is represented by enough sample points in time. This means that the required transient recorder would have to sample at very high rates. This is hard to achieve with ordinary electronic transient recorders of reasonable dynamic range. Secondly, the light available may simply be too weak to sample an analog intensity decay. Indeed the signal may consist of just single photons per excitation / emission. This is typically the case for single molecule experiments or work with minute sample volumes / concentrations. Then the discrete nature of the signal itself prohibits analog sampling. Even if one has more than just a single molecule and some reserve to increase the excitation power to obtain more fluorescence light, there will be limits, e.g. due to collection optic losses, spectral limits of detector sensitivity or photo–bleaching at higher excitation power. The solution is Time–Correlated Single Photon Counting (TCSPC). By using periodic excitation (typically from a laser) it is possible to extend the data collection over multiple excitation/emission cycles and one can then reconstruct the single cycle decay profile from single photon events collected over many cycles.

The TCSPC method is based on the repetitive, precisely timed registration of single photons of e.g., a fluorescence signal. The reference for the timing is the corresponding excitation pulse. A single photon detector such as a Photo Multiplier Tube (PMT) or a Single Photon Avalanche Photodiode (SPAD) is used to capture the fluorescence photons. Provided that the probability of registering more than one photon per cycle is low, the histogram of photon arrivals per time bin represents the time decay one would have obtained from a single shot time–resolved analog recording. The precondition of single photon probability can (and must) be met by attenuating the light level reaching the sample if necessary. If the single photon probability condition is met, there will actually be no photons registered in many of the excitation cycles. The diagrams below illustrate how the histogram is formed over multiple cycles.



The histogram is collected in a block of memory, where one memory cell holds the photon counts for one corresponding time bin. These time bins are often (historically) referred to as “time channels”. In practice, the registration of one photon involves the following steps: first, the time difference between the photon event and the corresponding excitation pulse must be measured. For this purpose both optical signals are converted to electrical signals. For the fluorescence photon this is done via the single photon detector mentioned previously. For the excitation pulse it may be done via another detector if there is no electrical sync signal supplied by the laser directly. Obviously, all conversion to electrical pulses must preserve the precise timing of the signals as accurately as possible. The actual time difference measurement is done by means of fast electronics which provide a digital timing result. This digital timing result is then used to address the histogram memory so that each possible timing value corresponds to one memory cell or histogram channel. Finally the addressed histogram cell is incremented. All steps are carried out by fast electronics so that the processing time required for each photon event is as short as possible. When sufficient counts have been collected, the histogram memory can be read out. The histogram data can then be used for display and e.g., fluorescence lifetime calculation. In the following we will expand on the various steps involved in the method and related issues of importance.

3.1. Count Rates and Single Photon Statistics

It was mentioned that in the past it was necessary to maintain a low probability of registering more than one photon per cycle. This condition ensured that the histogram of photon arrivals represents the time decay one would have obtained from a single shot time–resolved analog recording. The reason for this is briefly the following: Due to dead times of detector and electronics lasting at least some tens of nanoseconds after a photon event, historical TCSPC systems could register only one photon per excitation / emission cycle. If then the number of photons occurring in one excitation cycle were typically > 1, the system would very often register the first photon but miss the following one or more. This would lead to an over-representation of early photons in the histogram, an effect called ‘pile-up’. This leads to distortions of the fluorescence decay, typically making the fluorescence lifetime appear shorter. It was therefore crucial to keep the probability of cycles with more than one detected photon sufficiently low, unless one tolerates some error and/or corrects for it in data analysis.

To quantify the “safe” count rate limit for traditional TCSPC, one had to set acceptable error limits and apply some mathematical statistics. For practical purposes one would apply a rule of thumb: In order to maintain single photon statistics, on average only one in 20 to 100 excitation pulses should generate a count at the detector. In other words: the average count rate at the detector should be at most 1 % to 5 % of the excitation rate. Using e.g., a diode laser from PicoQuant's PDL Series, pulsed at 80 MHz repetition rate, the average detector count rate would then not exceed 4 Mcps. Typically one wants high count rates, in order to acquire fluorescence decay histograms quickly. This is of particular importance where dynamic lifetime changes or fast molecule transitions are to be studied or where large numbers of lifetime samples must be collected (e.g., in image scanning). This is why high laser rates (such as 40 or 80 MHz for the PDL Series) are important. PMTs can safely handle TCSPC count rates of over 10 Mcps. Most actively quenched SPADs may also operate up to 10 Mcps but some types suffer resolution degradation when operated too fast. Old NIM based TCSPC electronics could only handle a maximum of 50 to 500 kcps, while newer integrated TCSPC electronics typically reach count rates of 5 to 40 Mcps. With the HydraHarp 500 and modern Hybrid Photodetectors (HPD) count rates up to 85 Mcps can be collected.

It is worth noting that the photon arrival times are typically random so that there can be bursts of high count rate and periods of low count rates. Bursts may exceed the average rate. The average (sustained) rate the instrument can deal with is important when losses cannot be tolerated, notably in time tagging applications. Due to the high bandwidth of USB 3.0 the HydraHarp 500 can handle sustained time tagging rates as high as 85 Mcps through this interface. The external FPGA interface can even handle even higher rates. When comparing count rates considered here and elsewhere please pay attention to the details. The specifications for TCSPC systems often interpret their maximum count rates differently. This is why in this context dead time is also of interest. It describes the time the system cannot register photons while it is processing a previous photon event. The term is applicable both to detectors and electronics. The HydraHarp 500 has an extremely short dead time of about 680 ps, imposing the smallest losses among comparable instruments today. The short dead time of the HydraHarp 500 together with its multi-stop capability now allows measurement scenarios where the classic pile-up limit is no longer critical. This is of special interest for very fast FLIM measurements and PicoQuant's concept of rapidFLIM exploits the idea. Using an HPD (e.g., PicoQuant PMA Hybrid) that permits count rates as high as 80 Mcps one can obtain FLIM images at unprecedented speed. While classic pile-up due to dead time is no longer an issue, one must now consider the effect of pulse-pile-up. This effect occurs when photons arrive at the detector with very short temporal spacing, such that the detector can no longer produce separate output pulses. This merging of detector pulses leads to another type of artifact in the decay histograms. A numerical correction of these artifacts during data analysis ensures that lifetimes and amplitudes are still obtained correctly for very fast quantitative measurements. See section 3.5 or www.picoquant.com for related publications.

3.2. Timing Resolution

The most critical component for the timing resolution in TCSPC measurements is usually the detector. However, in contrast to analog transient recording, the time resolution of TCSPC is not limited by the impulse response of the detector. Only the timing precision of registering a photon determines the resolution. This is limited by the timing uncertainty that the detector introduces in the conversion from a photon to an electrical pulse. This timing error or uncertainty can be as much as ten times smaller than the detector's pulse response. The timing uncertainties are usually quantified by specifying the rms error (standard deviation) or the Full Width at Half Maximum (FWHM) of the timing distribution or instrument response function (IRF). Note that these two notations are related but not identical[1]. Micro channel plate PMTs, can achieve timing uncertainties as small as 25 ps FWHM. Lower cost PMTs or SPADs may introduce uncertainties of 50 to 500 ps FWHM, HPDs lie in between with typical uncertainties of 50..150 ps FWHM. Superconducting nanowire detectors have timing uncertainties of typically 20 to 100 ps FWHM, some optimized designs can even reach below 10 ps.

The second most critical source of IRF broadening in fluorescence lifetime measurements with TCSPC is usually the excitation source. While many lasers can provide sufficiently short pulses, it is also necessary to obtain an electrical timing reference signal (sync) for comparison with the fluorescence photon signal. The type of sync signal that is available depends on the excitation source. With gain switched diode lasers (e.g., PDL 800–D) a low jitter electrical sync signal is readily available. The sync signal used here is typically a narrow negative pulse of −800 mV into 50 Ω (NIM standard). The sharp falling edge is synchronous with the laser pulse (< 3 ps rms jitter for the PDL 800–D). With other lasers (e.g., Ti:Sa) a second detector must be used to derive a sync signal from the optical pulse train. This is commonly done with a fast photo diode (APD or PIN diode). The light for this reference detector must be derived from the excitation laser beam e.g., by means of a semi–transparent mirror. The reference detector must be chosen and set up carefully as it contributes to the overall timing error.

Another source of timing error is the timing jitter of the electronic components used for TCSPC. This is caused by the finite rise / fall–time of the electrical signals used for the time measurement. At the trigger point of comparators, logic gates etc., the amplitude noise (thermal noise, interference etc.) always present in these signals is transformed to a corresponding timing error (phase noise). However, the contribution of the electronics to the total timing error is usually very small.

Generally, it is always a good idea to keep electrical noise pick-up low in all system components. Uncorrelated electrical noise will cause just random jitter and IRF broadening but correlated noise can cause even more disturbing artifacts. This is why signal leads should be properly shielded coax cables, and strong sources of electromagnetic interference should be kept away from the TCSPC detector and electronics.

The contribution of the time spread introduced by the individual components of a TCSPC system to the total IRF strongly depends on their relative magnitude. Strictly speaking, the overall IRF is the convolution of all component IRFs. An estimate of the overall IRF width, assuming independent noise sources, can be obtained from the geometric sum of the individual components as an rms figure according to statistical error propagation laws:

$$e_{system} = \sqrt{\sum e_{component}^2}$$

As the individual contributions are squared, the total will be dominated by the largest component. It is therefore of little value to improve a component that is already relatively good. If e.g., the detector has an IRF width of 200 ps FWHM, shortening the laser pulse from 50 ps to 40 ps will have practically no effect. Although it is difficult to specify a general lower limit on the fluorescence lifetime that can be measured by a given TCSPC instrument, as a rule of thumb, one can assume that under favorable conditions lifetimes down to 1 / 10 of the IRF width (FWHM) can still be recovered via deconvolution.

A final time–resolution related issue worth noting here is the bin width of the TCSPC histogram. As outlined above, the analog electronic processing of the timing signals (detector, amplifiers, etc.) creates a continuous distribution around any true time value. In order to form a histogram, at some point the timing results must be quantized. This quantization introduces further error, if chosen too coarse. The quantization step width (i.e. the resolution) should therefore be small compared to the IRF width. As a minimum sampling frequency, from the point of view of information theory, one would assume the Nyquist frequency. That is, the signal should be sampled at least at twice the highest frequency contained in it. For practical purposes one may wish to exceed this limit where possible, but there is usually little benefit in sampling the histogram at resolutions higher than 1 / 10 of the overall IRF width of the analog part of the system.

1 In case of a Gaussian distribution the r.m.s deviation corresponds to the standard deviation σ and FWHM ≈ 2,35 σ.

3.3. Photon Counting Detectors

3.3.1. Photomultiplier Tube (PMT)

A PMT consists of a light–sensitive photo cathode that generates electrons when exposed to light. These electrons are directed onto a charged electrode called dynode. The collision of the electrons with the dynode produces additional electrons. Since each electron that strikes the dynode causes several electrons to be emitted, there is a multiplication effect. After further amplification by multiple dynodes, the electrons are collected at the anode of the PMT and output as a current. The current is directly proportional to the light intensity striking the photo cathode. Because of the multiplicative effect of the dynode chain, the PMT is a photo electron amplifier with high sensitivity and remarkably low noise. The high voltage driving the tube may be varied to change the sensitivity of the PMT. Current PMTs have a wide dynamic range, i.e. they can also measure relatively high levels of light. Furthermore, they are very fast, so that rapid successive events can be reliably monitored. One photon on the photo cathode can produce a short output pulse containing millions of photoelectrons. PMTs can therefore be used as single photon detectors. In photon counting mode, individual photons that strike the photo cathode of the PMT are registered. Each photon event gives rise to an electrical pulse at the output. The number of pulses, or counts per second, is proportional to the light impinging upon the PMT. As the number of photon events increase at higher light levels, it will become difficult to differentiate between individual pulses and the photon counting detector's behavior will become non–linear. This usually occurs between 1 and 20 Mcps, depending on the detector design. Similarly, in TCSPC applications, individual photon pulses may merge into one as the count rate increases. This leads to pulse pile–up and distortions of the collected histograms.

The timing uncertainty between photon arrival and electrical output (transit time spread) is usually small enough to permit time–resolved photon counting at a sub–nanosecond scale. In single photon counting mode the tube is typically operated at a constant high voltage where the PMT is most sensitive.

PMTs usually operate within the blue to red regions of the visible spectrum, with greatest quantum efficiency in the blue–green region, depending upon photo–cathode materials. Typical quantum efficiencies are about 25 %. For spectroscopy experiments in the ultraviolet / visible / near infrared region of the spectrum, a PMT is very well suited.

Because of noise from various sources in the tube, the output of the PMT may contain pulses that are not related to the light input. These are referred to as dark counts. The detection system can to some extent reject these spurious pulses by means of electronic discriminator circuitry. This discrimination is based on the probability that some of the noise generated pulses (those from the dynodes) exhibit lower signal levels than pulses from a true photon event. Thermal emission from the cathode that undergoes the full amplification process can usually not be suppressed this way. In this case cooling of the detector is more helpful.

3.3.2. Micro Channel Plate PMT (MCP)

A Micro Channel Plate PMT consists of an array of glass capillaries (5–25 µm inner diameter) whose insides are coated with an electron–emissive material. The capillaries are biased at a high voltage. Like in a PMT, an electron that strikes the inside wall of one of the capillaries creates an avalanche of secondary electrons. This cascading effect creates a gain of 10^3 to 10^6 and produces a current pulse at the output. Due to the narrow and well defined electron path inside the capillaries, the transit time spread of the output pulses is much reduced compared to a normal PMT. The timing jitter of MCPs is therefore sufficiently small to perform time–resolved photon counting on a picosecond scale, usually outperforming PMTs. Good but also expensive MCPs can achieve timing uncertainties as low as 25 ps. In this respect, microchannel plates are a good match for the HydraHarp 500 but they are quite limited in permitted count rate and provide lower sensitivity towards the red end of the spectrum compared to suitably optimized SPADs.

3.3.3. Single Photon Avalanche Photo Diode (SPAD)

Avalanche Photo Diodes (APDs) are semiconductor devices, usually restricted to operation in the visible to infrared parts of the spectrum. Generally, APDs may be used for ultra–low light detection (optical powers < 1 pW), and can also be used as photon–counters in the so–called "Geiger" mode (biased slightly above the breakdown voltage). In the latter case, a single photon may trigger an avalanche of about 10^8 carriers. In this mode the device can be used as a detector for photon counting with very accurate timing of the photon arrival. In this context they are also referred to as Single Photon Avalanche Photo Diodes (SPAD). Selected devices with small active surfaces may achieve timing accuracies down to 30 ps, but are usually hard to align and difficult to focus into. SPADs are sometimes noisier than PMTs, but can have a greater quantum efficiency especially towards the red side of the spectrum. Maximum quantum efficiencies are about 70 %. Such sensitive devices provide a timing precision of ~300..400 ps. Commercial modules are thermoelectrically cooled for low dark count rate and deliver pre–shaped TTL pulses. They are the most common detectors for applications where NIR sensitivity is important, e.g., single molecule detection. To achieve the specified timing precision, exact focusing into the center of the active area is necessary. Other SPAD designs such as the PDM family from Micro Photon Devices have the benefit of much better timing resolution and robustness, however, at the expense of a lower sensitivity at the red end of the spectrum.

3.3.4. Other and Novel Photon Detectors

The field of photon detectors is still evolving. Recent developments include so called silicon PMTs, Hybrid Photo Detectors (HPDs), superconducting nanowire detectors and APDs with sufficient gain for single photon detection in analog mode. Each of these detectors have their specific benefits and shortcomings. Only a very brief overview will be given here.

Silicon PMTs are essentially arrays of SPADs, all coupled to a common output. This has the benefit of creating a large area detector that can even resolve photon numbers. The drawback is increased dark count rate and reduced timing precision.

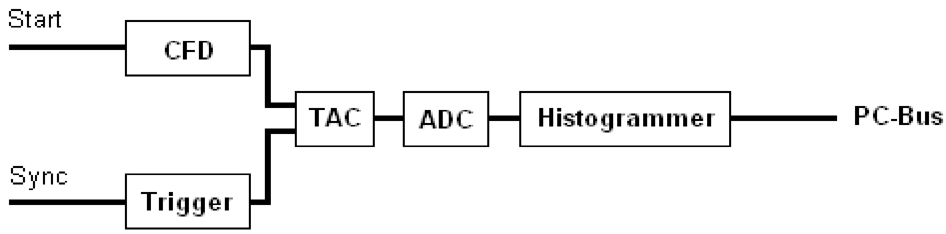
HPDs make use of a combination of a PMT-like front end followed by an APD structure. The benefits are good timing performance and virtually zero afterpulsing while the need for very high voltage is a disadvantage. PicoQuant's PMA Hybrid series include the high voltage in an easy-to-use package.

Superconducting nanowires single photon detectors (SNSPDs or SSPDs) routinely achieve excellent timing performance (<30 ps jitter) and high sensitivity from the visible to the near infrared with overall system detection efficiencies in excess of 90%. They operate at cryogenic temperatures, typically between 0.8 K and 4 K, which incurs cost, large footprint and power consumption and makes them less practical for applications where these parameters are of concern.

Another class of potentially interesting detectors which have recently emerged, are APDs with very high gain. In combination with an electronic amplifier they have been shown to detect single photons. As opposed to Geiger mode, this avoids afterpulsing and allows for very fast counting rates. The disadvantage is a high dark count rate, currently too high for any practical TCSPC application.

3.4. Principles Behind the TCSPC Electronics

For introductory purposes it is worth to look first at the design of historical TCSPC systems. They consist of the following building blocks:



The Constant Fraction Discriminator (CFD) is used to extract precise timing information from the electrical detector pulses that may vary in amplitude. This way the overall system IRF may be tuned to become narrower and some of the random background signal can be suppressed. The same could not be achieved with a simple edge+level trigger (comparator). Especially with PMTs, constant fraction discrimination is important, because their pulse amplitudes vary significantly. In particular pulses originating from random electrons generated at the dynodes of the PMT can be suppressed because their avalanches had less time to amplify and their corresponding output pulses are small.

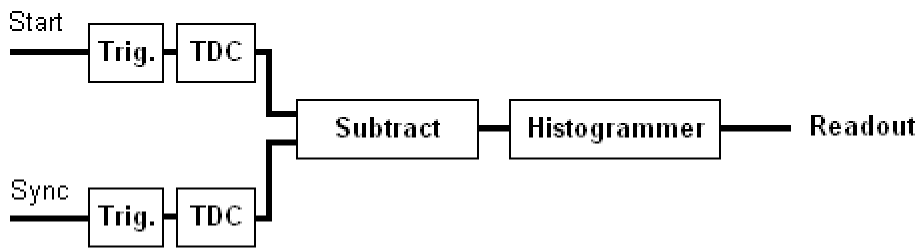
The principle of a classic CFD is the comparison of the original detector signal with an amplified, inverted and delayed version of itself. The signal derived from this comparison changes its polarity exactly when a constant fraction of the detector pulse height is reached. The zero crossing point of this signal is therefore suitable to derive a timing signal independent from the amplitude of the input pulse. In practice the comparison is done by a summation. The timing is done by a subsequent threshold trigger of the sum signal using a settable level, the so called zero cross trigger. Newer CFD designs achieve the same objective by differentiating the input signal and triggering on the zero crossing of the differentiated signal. This has the benefit of adapting to different detector types without a need for changing physical delay lines.

It must be noted that because of its delay element a CFD requires time to make its “decision” and this prolongs dead time. Since short dead time is a precious feature when using high speed detectors, the use of a CFD may harm. A simple settable comparator (edge trigger) is possibly preferred then. Indeed many modern detectors have very steep signal edges that do not require a CFD. In similarity to the detector signal, the sync signal must be made available to the timing circuitry. Since the sync pulses are usually of well defined amplitude and shape, the edge trigger is sufficient to accommodate different sync sources.

In historical TCSPC systems the signals from the two input discriminators / triggers are fed to a Time to Amplitude Converter (TAC). This circuit is essentially a highly linear ramp generator that is started by one signal and stopped by the other. The result is a voltage proportional to the time difference between the two signals. In such conventional systems the voltage obtained from the TAC is then fed to an Analog to Digital Converter (ADC) which provides the digital timing value used to address the histogrammer. The ADC must be very fast in order to keep the dead time of the system short. Furthermore it must guarantee a very good linearity (both over the full range as well as differentially). These are criteria difficult to meet simultaneously, particularly with ADCs of high resolution (e.g. 12 bits) as is desirable for TCSPC over many time bins.

The histogrammer has to increment each histogram memory cell, whose digital address in the histogram memory it receives from the ADC. This is commonly done by fast digital logic e.g., in the form of Field Programmable Gate Arrays (FPGA) or a microprocessor.

While this section so far outlined the typical structure of conventional TCSPC systems, it is important to note that the design of the HydraHarp 500 is different. Today, it is state–of–the–art that the tasks conventionally performed by TAC and ADC are carried out by a so called Time to Digital Converter (TDC). These circuits allow not only picosecond timing but can also extend the measurable time span to virtually any length by means of digital counters. The HydraHarp 500 uses one such circuit in each input channel and one for the sync input. They independently work on each input signal and provide picosecond arrival times that then can be processed further, with a lot more options than in conventional TCSPC systems. In the case of classic TCSPC, this processing consists of a subtraction of the two time figures and histogramming of the differences. This is identical to the classic start–stop measurements of the conventional TAC approach. The following figure exemplifies this for one detector channel (Start).



The full strength of the HydraHarp 500 design is exploited by collecting the unprocessed independent arrival times as a continuous data stream for more advanced analysis. Details on such advanced analysis can be found in the literature (see section 3.5). In this case the on–board memory is reconfigured as a large data buffer (First In, First Out; FIFO) so that count rate bursts and irregular data transfer are decoupled. This permits uninterrupted continuous data collection with high throughput. This mode of operation is called Time–Tagged Time–Resolved (TTTR) mode or just “time tagging”. Details can be found in section 6.3.

Forward and Reverse Start–Stop Mode

For simplicity it is most convenient to assume that the time delay measurement is directly causal, i.e. the laser pulse causes a photon event and one measures the time delay between laser pulse and the subsequent photon event. However, most conventional TCSPC systems needed to give up this logical concept because of the typically high repetition rates of the excitation lasers: Since the time measurement circuit cannot know in advance whether there will be a fluorescence photon, it would have to start a time measurement upon each laser pulse. Considering that conventional TAC conversion times were in the region of .1 to 2 μs, any excitation rate in excess of .5 to 10 MHz would overrun the time measurement circuits. In fact they would most of the time be busy with conversions that never complete, because there is no photon event at all in most cycles. By reversing the start and stop signals in the time measurement, the conversion rates are only as high as the actual photon rates generated by the fluorescent sample. Historically there were (and had to be) only about 1 to 5 % of the excitation rate and could therefore be handled easily by the TAC / ADC. The consequence of this approach, however, is that the times measured are not those between laser pulse and corresponding photon event, but those between photon event and the next laser pulse (unless a long cable delay is inserted). This still works (by software data reversing) but is inconvenient in two ways:

1) Having to reverse the data leads to unpleasant relocation of the data displayed on a true time axis when the time resolution is changed.
2) Changing between slow and fast excitation sources requires reconnecting to different inputs.

The HydraHarp 500 is very different in this respect, as it allows to work in forward start stop mode, even with very fast lasers. This is facilitated by two design features:

1) Independent operation of the TDCs of all channels, and
2) a programmable divider in front of the sync input.

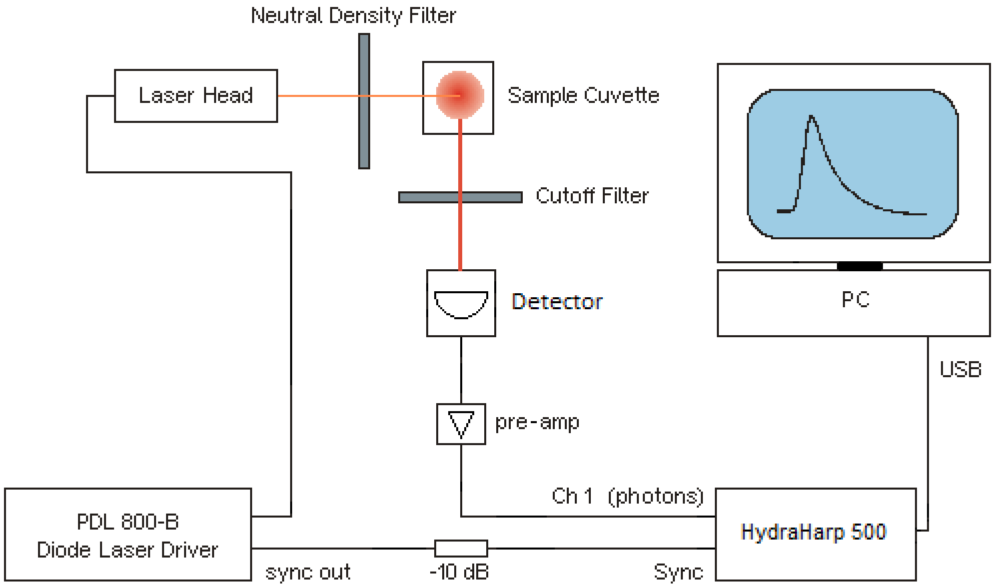
The divider allows to reduce the sync input rate so that the front end processing can keep up with it. Internal logic determines the sync period and re-calculates the sync signals that were divided out. Sync rates as high as 1.2 GHz can thereby be handled. It should be noted that this only works with stable sync sources that provide a constant pulse-to-pulse period. Virtually all currently available fast laser sources meet this requirement within an error band of a few picoseconds. Note: for slow sync sources (< 1 MHz) the sync divider should not be used (set to None). Similarly, the divider must not be used in coincidence correlation measurements (or similar applications) when the sync input receives non-periodic (random) pulses from a photon detector. In summary: The HydraHarp 500 is designed to always work in forward start-stop mode.

Experimental Setup for Fluorescence Decay Measurements with TCSPC

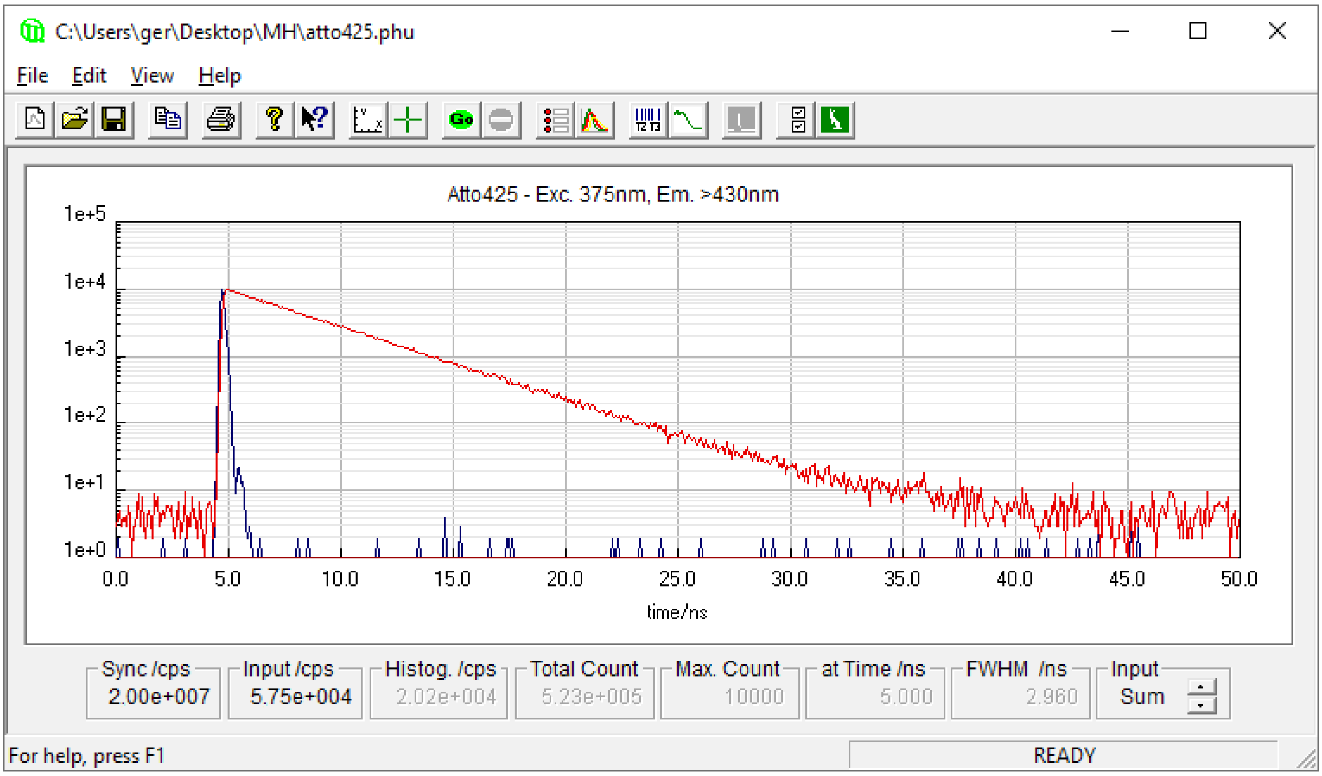
The next figure below shows a typical setup for fluorescence lifetime measurements using one input channel of the HydraHarp 500.

The picosecond diode laser (PDL 800–B driver with attached laser head) is triggered by its internal oscillator (settable at 2.5, 5, 10, 20 and 40 MHz). The light pulses of typically <70 ps FWHM, are directed toward the sample cuvette via appropriate optics. A neutral density filter can be used to attenuate the light levels if necessary. Upon excitation, the fluorescent sample will emit light at a longer wavelength than that of the excitation light. The fluorescence light is filtered out from scattered excitation light by means of an optical cut-off filter (other configurations may use a monochromator here). Then it is directed to the photon detector, again possibly via some appropriate collection optics, e.g., a microscope objective or a lens.

As a photon detector the 10721 PMTs from Hamamatsu are very convenient. They only need a 5 V supply and have an instrument response width of <250 ps, allowing measuring lifetimes even much shorter than this via deconvolution. If a higher time resolution is required, the detector of choice is an MCP-PMT or a HPD. The electrical signal obtained from the detector (small negative pulses of typically −10 to −50 mV) is fed to the TCSPC electronics via a preamplifier (e.g., PAM 105 from PicoQuant). This gives pulses of −100 to −500 mV. The trigger level is then set for e.g. -80 mV. Cabling is double shielded 50 Ω coax cable. If the detector is a SPAD module with TTL output then the input trigger level must be set to a positive voltage and rising edge. TTL signals must be be attenuated (-10 dB) to avoid input damage and to reduce cross-talk. Some SPAD devices like the PDM series from MPD provide negative timing signals that can also be slightly attenuated (-10 dB) for lower cross-talk. The PDL Series of laser drivers readily provide the electric sync signal needed for the photon arrival time measurement. This signal (a narrow negative pulse) is also fed to the TCSPC electronics via a high quality 50 Ω coax cable. When using the HydraHarp 500 in combination with the PDL 800–B or a similar PicoQuant laser driver, a 10 dB attenuator can be inserted directly at the sync output of the laser driver. This reduces cross-talk. when the detector signals are relatively weak. If the laser does not provide an electrical sync signal (e.g. Ti:Sa lasers), a sync detector (photo diode) such as the TDA 200 must be used.



The following figure shows TCSPC histograms obtained with this kind of setup. Excitation source was a PDL 800–B (PicoQuant) with a 375 nm laser head running at 20 MHz repetition rate. Detector was a PMA Hybrid 06 from PicoQuant with internal amplifier. The cut-off filter was a longpass for >430nm. The narrower peak (blue curve) represents the system IRF, here dominated by laser and detector. The other curve (red) is the fluorescence decay from a solution of ATTO 425 in water, a fluorescent dye with fairly short fluorescence lifetime (~3.7 ns). The count rate was adjusted to <3% of the laser rate to safely prevent pile-up. The plot in logarithmic scale shows the perfect mono-exponential nature of the decay curve, as one would expect. Note that this is obtained without a deconvolution of the IRF.



The approximate mono–exponential fluorescence lifetime can be obtained from a simple comparison of two points in the curve with counts in the ratio of $1 : {}^{1}/_{e}$ (e.g. 100 000 : 36 788). For a precise measurement one would perform a numerical exponential fit with IRF deconvolution (typically implemented as an iterative reconvolution). This would result in slightly shorter lifetimes since the IRF broadens the decay. Indeed one can measure lifetimes significantly smaller than the IRF with this method. Additionally, the residues of the fit can then be used to assess the quality of the fit and thereby the reliability of the lifetime measurement. The "EasyTau 2" software package from PicoQuant provides this functionality.

3.5. Further Reading

1. O'Connor, D.V.O., Phillips, D.:
   Time–correlated Single Photon Counting, Academic Press, London, 1984

2. Lakowicz, J. R.:
   Principles of Fluorescence Spectroscopy, 3rd Edition, Springer, New York, 2006

3. Kapusta, P., Wahl, M., Erdmann, R. (Eds.):
   Advanced Photon Counting - Applications, Methods, Instrumentation
   Springer Series on Fluorescence, Vol. 15, Springer International Publishing, 2015
   ISBN 978-3-319-15635-4

4. Ortmann U., Wahl M., Kapusta P.:
   Time-resolved fluorescence: Novel technical solutions.
   Springer Series on Fluorescence, Vol. 5, p.259-275, Springer International Publishing, 2008

5. Wahl M., Roehlicke T., Kulisch S., Rohilla S., Kraemer B., Hocke A.C.:
   Photon arrival time tagging with many channels, sub-nanosecond deadtime, very high throughput, and fiber optic remote synchronization. *Review of Scientific Instruments*, 91, 013108 (2020)
   preprint available on ArXiv

6. Terhaar R., Rödiger J., Häußler M., Wahl M., Gehring H., Wolff M.A., Beutel F., Hartmann W., Walter N., Hanke J., Hanne P., Walenta N., Diedrich M., Perlot N., Tillmann N., Röhlicke T., Ahangarianabhari M., Schuck C., and Pernice W.H.P. : Ultrafast quantum key distribution using fully parallelized quantum channels. *Optics Express*, Vol. 31, Issue 2, pp. 2675-2688 (2023)
   preprint available on ArXiv

7. Wahl M., Rahn H.-J., Röhlicke T., Erdmann , R., Kell G., Ahlrichs, A., Kernbach, M., Schell, A.W., Benson, O.: Integrated Multichannel Photon Timing Instrument with Very Short Dead Time and High Throughput. *Review of Scientific Instruments*, 84, 043102 (2013)

8. Wahl M., Rahn H.-J., Röhlicke T., Kell G., Nettels D., Hillger F., Schuler B., Erdmann R.:
   Scalable time-correlated photon counting system with multiple independent input channels.
   Review of Scientific Instruments, Vol.79, 123113 (2008)

9. Koberling F., Kraemer B., Buschmann V., Ruettinger S., Kapusta P., Patting M., Wahl M., Erdmann R.:
   Recent advances in photon coincidence measurements for photon antibunching and full correlation analysis. *Proceedings of SPIE*, Vol.7185, 71850Q (2009)

10. O'Connor, D.V.O., Ware, W.R., Andre, J.C.:
    Deconvolution of fluorescence decay curves. A critical comparison of techniques.
    J. Phys. Chem. 83, 1333–1343, 1979

11. Patting M., Wahl M., Kapusta P., Erdmann R.:
    Dead-time effects in TCSPC data analysis. *Proceedings of SPIE*, Vol.6583, 658307 (2007)

12. Patting M., Reisch, P., Sackrow, M., Dowler, R., Koenig, M., Wahl M.:
    Fluorescence decay data analysis correcting for detector pulse pile-up at very high count rates.
    Optical Engineering, 57(3), 031305 (2018). doi: 10.1117/1.OE.57.3.031305

13. Bibliography listing all publications with work based on PicoQuant instruments
    http://www.picoquant.com/scientific/references

14. Web page of the PicoQuant TCSPC and time tagging devices
    https://www.picoquant.com/products/category/tcspc-and-time-tagging-modules

15. PicoQuant technical and application notes
    http://www.picoquant.com/scientific/technical-and-application-notes

4. Hardware and Software Installation

4.1. Scope

This chapter covers the hardware installation of the HydraHarp 500. Regarding software this manual describes solely **the standard HydraHarp 500 software** providing users of the instruments with an easy to use graphical interface. The HydraHarp 500 software runs on current x86-64 Windows PC platforms and also under Linux with Wine (see section 9.4). It provides functions for setting measurement parameters, displaying measurement results, loading and saving of measurement parameters and decay curves, etc.

In addition to the standard HydraHarp 500 software there are a variety of other software **items that are not covered by this manual**:

HH500Lib is a programming library enabling users to write custom data acquisition programs for the HydraHarp 500 in virtually all popular programming environments for Microsoft Windows. There is also a version for Linux (AMD/Intel x64 processor architecture only) which is fully compatible with that for Windows so that applications can easily be ported across the two operating systems. The programming library is not covered here in this manual. Please see the separate installation and documentation files (separate for Windows and Linux) provided as part of the distribution download from the HydraHarp 500 product pages on the Web.

The external FPGA interface (EFI) permits retrieving TTTR mode data at substantially higher bandwidth and lower latency than via USB. This enables custom data processing in real-time, way beyond the capabilities of a PC in terms of speed and latency. However, since the EFI is an expert tool requiring knowledge of FPGA programming and writing suitable custom software, it is not supported by the standard HydraHarp 500 software described here in this manual. In order to enable and use the EFI from the software side there are a set of dedicated routines provided in the programming library mentioned above. Since using the EFI is an advanced topic in its own, also involving a fair amount of FPGA programming details, there is a separate manual for this. It is provided as part of the EFI gateware and software pack which can be downloaded from the HydraHarp 500 product pages on the Web.

A relatively advanced high-level API package “snAPI” for Python is also available (Windows only). It readily provides data collection and file writing methods as well as many real-time analysis methods such as intensity and coincidence time traces, FCS and $g^{(2)}$ correlation.

Further items of PicoQuant software supporting the HydraHarp 500 devices are the UniHarp software, the EasyTau 2 fluorescence decay fit software, the SymPhoTime 64 microscopy software, and the QuCoa software for quantum optics applications. These allow highly professional, application focused use of the devices in the contexts of time-resolved spectroscopy, microscopy, and quantum optical research. Please see the PicoQuant website for more information on these items.

4.2. What's New in this Version

The HydraHarp 500 is a brand new product and the HydraHarp 500 software version 1.0 is the first release. Please note that by name and specifications the HydraHarp 500 is a modern successor of the seminal HydraHarp 400, however, they do not share the same software. Make sure you download and use only the latest suitable software for the particular hardware model you have.

4.3. General Installation Notes

When handling the HydraHarp 500 system, make sure to avoid electrostatic discharges, especially when connecting cables. Before connecting any signals, carefully study the maximum ratings given in section 9.3.1.

Note that the other PicoQuant TCSPC devices have their own software and their own manuals. The HydraHarp 500 software described here does not work with these devices and this manual does not relate to them in a meaningful way.

The HydraHarp 500 software version 1.0.0.0 is compatible with x64 Windows[1] 10 and 11 as of August 2025. Windows 7 and 8 may still work but are no longer officially supported. Consider the security risks of using an outdated operating system that no longer receives updates. Future Windows versions are likely to work but obviously cannot be tested before they are released, and are therefore not formally supported.

1 Trademark of Microsoft Inc.

In order to use the HydraHarp 500 software with the HydraHarp 500 hardware, a compatible device driver must be installed. The HydraHarp 500 is a ‘plug and play’ device, meaning that necessary resources and drivers are allocated automatically by the operating system. The software setup will conveniently cater for installing the driver.

On some Windows installations you may need administrator status to perform software setup and de–installation. For the driver installation, it is needed in any case. Installing as administrator has the benefit that you can install the software for use by all users on that computer even if they have limited access rights. The HydraHarp 500 software will maintain individual settings for each user and each device in the Windows registry. Note that the HydraHarp 500 software does not support running in multiple concurrent user sessions.

4.4. Software Installation

Please download the HydraHarp 500 software pack from the PicoQuant website at https://www.picoquant.com/downloads. On the download page scroll down to the category `TCSPC and Time Tagging Electronics` and select `HydraHarp 500 – Operation Software`. The download will come as a ZIP-File. Please unzip this file to a temporary hard disk location of your choice. You will need administrator status to perform the setup for all users. For the device driver installation you need it in any case.

For access to the HydraHarp 500 hardware, a device driver must be installed. The HydraHarp 500 is a ‘plug and play’ device, this means that the necessary resources and drivers are allocated automatically by the operating system. Windows will recognize when such a device is connected and tries to install the appropriate driver. It is recommended to perform the software setup before connecting the device because then Windows will find the driver readily installed. If you skipped that step, Windows may report that it could not install the driver. In that case you can still run the software setup as described below. Alternatively you can direct Windows to search the location where you unpacked the setup files.

To perform the software installation open the disk location where you unpacked the setup files.
The installer program file containing the complete distribution is named `setup.exe` .
Run `setup.exe`. The setup program will guide you through the installation process step by step.


Setup

When asked for a destination folder for the new software, please accept the default path or select another according to your program storage policies. This is where the HydraHarp 500 application files will be installed. To avoid confusion, make sure not to specify the path of an older HydraHarp 500 software version that you may not have uninstalled or that of any other program on your PC.

The default location is: `\Program Files\PicoQuant\HydraHarp500v10`.

Setup will also create a dedicated "program folder" for the new HydraHarp 500 software that will later appear in the Windows Start Menu. You can accept the default folder name or select another according to your own naming policies. However, you should make sure not to specify the folder name of an older software version that you have not uninstalled nor the dedicated folder of any other program.

In the chosen destination folder the installer will also create a subdirectory `\filedemo` which contains demo source code for access to HydraHarp 500 data files in various programming languages. Furthermore, another subdirectory `\sampledata` will be created with samples of HydraHarp 500 data files. Other necessary files such as setup information and the device driver will be installed in the standard places in your Windows directory tree.

The setup program will also optionally install the device driver and a *File Info* shell extension that you can use to inspect individual header items of a `*.ptu` or `*.phu` file. It also shows the measurement mode. Just right-click on the file in Windows explorer and select *Properties*. Then look at the tab *PQ File Info* and the tab *PQ File Comment*.

After the installation the HydraHarp 500 software should be available in the Windows Start Menu under the folder name you chose during setup. If you accepted the default then it will appear under
Programs | PicoQuant – HydraHarp 500 v1.0.

4.5. Hardware Installation

Make sure to *prevent electrostatic discharge* when unpacking and handling your HydraHarp 500, especially when connecting cables. Note that PMT detectors operate with high voltage that may discharge through the signal cable. Make sure such detectors are switched off and fully discharged before connecting them.

All HydraHarp 500 hardware models receive their power from a regular 100...240V AC line. The corresponding IEC socket at the back of the housing also holds the power switch and two fuses. The HydraHarp 500 has no dedicated power indicator, however, once power is connected and switched on you should see the three status LEDs at the front light up. See section 9.3.3 for their interpretation.

Also at the back of the housing there is a grounding terminal that can be used to make a connection to the grounding line of your lab. Ask a qualified electrician when in doubt what and where this is. This grounding is not necessary for electrical safety, however, it may in some situations help to reduce noise pickup. In order to be effective the grounding should be of low impedance, i.e. using thick wires, less than 3 m long, ideally as short as possible.

The HydraHarp 500 has a USB 3.0 super speed interface which is backward compatible with USB 2.0. In order to obtain the maximum throughput it must be connected through USB 3.0 or higher. Consult your PC manual as to whether and where it provides high speed USB 2.0 or super speed USB 3.x connectors. The latter can usually be identified by their blue color. The HydraHarp 500 will not work with a USB 1.x connection. However, all current PCs should readily provide USB 3.0 connectivity. Recent PCs provide USB 3.1 or 3.2 connections which are backward compatible with USB 3.0 and should be fine for running the HydraHarp 500. If the PC has no USB 3.x ports you can install a USB 3.x adapter card but this may not provide the same throughput as an on-board controller.

Always use a quality USB cable rated for USB 3.0 or higher speed. The cable length must not exceed 5 meters (~16 ft). For best reliability we recommended to use cables of max. 3 meters length, preferrably the one that shipped with your HydraHarp 500. Note that the USB specification does not allow cable extensions other than dedicated active extension cables or hubs. The HydraHarp 500 should work flawlessly through suitable USB hubs. This is also a valid way of extending the maximum cable length. After a hub, another cable of up to 5 meters is allowed. Note, however, that hubs may lower the data throughput. For the same reasons it is recommended not to connect other devices with high bandwidth requirements (e.g. USB cameras) to the same host controller or hub. In order to conveniently find out how many USB controllers you have and which one the individual USB sockets belong to, you can use Microsoft's Universal Serial Bus Viewer (*USBView*) .

The steps below describe how to connect the signals for your experiment. The inputs for the photon detector(s) and the sync signal are SMA connectors located on the front panel of the HydraHarp 500. They are labeled SYNC and CH1...CHx with largest x dependent on the model. The sync input is typically used for synchronization with a laser and CH1..CHx are typically used for detector signals. The sync input can also be used for detectors if a laser sync is not required, e.g., in antibunching measurements or in generic time tagging applications. All inputs are terminated with 50 Ohms internally. Use quality 50 Ohms coax cables with appropriate connectors. For interfacing to BNC connectors, use standard 50 Ohms adapters. Carefully screw on the SMA connectors for sync and detector(s) until they are hand-tight. Do not use wrenches. Your HydraHarp 500 ships with a dedicated SMA connector tool that allows convenient handling in limited space without risk of over-tightening. Note that PMT detectors operate with high voltages that may discharge through the signal cable. Make sure such detectors are switched off and fully discharged before connecting or disconnecting them.

In case of time resolved fluorescence experiments with a pulsed excitation source, the sync signal must be connected to the sync input and the detector signal(s) must be connected to the detector input(s). If coincidence correlation experiments between two (or more) detector signals are to be carried out, you need to decide whether you will be using histogramming or TTTR modes T2/T3. In the case of histogramming and T3 mode connect one detector to the sync input and one or more to the detector inputs. Histogramming and T3 timing will always be with respect to the sync input. In T2 mode it is possible to determine the relative timing between all inputs but this requires dedicated software (see section 6.3).

Connect the other signal cable ends to the appropriate signal sources (50 Ω) in your experimental setup. The inputs of the HydraHarp 500 accept positive or negative pulses with peak values of up to +1.2 V or -1.2 V, respectively. All inputs should be operated with similar pulse amplitudes to minimize cross-talk. The optimum amplitude range is 100 to 200 mV. Below this range you may pick up noise, above there may be cross-talk. Most PMT and MCP detectors will require a pre–amplifier to reach enough signal level. TTL–SPAD–detectors must be connected through an attenuator or an attenuating inverter (PicoQuant SIA 400). **Never connect TTL signals directly, as this may cause damage.** Weak PMT detectors should be connected through a 20 dB high speed pre–amplifier. MCP–PMT detectors should be connected through an amplifier with slightly higher gain. When detectors with small signals are being used in combination with laser drivers from PicoQuant's PDL Series, the sync pulses from the laser driver should be attenuated by 10 to 15 dB to fall into the optimum range for smallest cross-talk. Similar attenuation is recommended on the detector signals when detectors with NIM output are used. Suitable attenuator and amplifier devices are available from PicoQuant.

IMPORTANT: switch the high voltage supply of PMTs off and allow their electrodes to discharge before connecting them. Their high voltage charge may damage the pre–amplifier. Observe the allowed maximum ratings for the input signal levels. Above these levels hardware damage will occur. If you are not sure what signals your setup delivers, use a fast oscilloscope to check the signal level and shape before connecting them to the HydraHarp 500 inputs. All signals should have fast rise times of no more than a few ns. Slower signals may degrade timing accuracy and precision.

Do not connect anything other than dedicated hardware to the Sub-D multi-pin connector (control port). It is provided for hardware expansion (notably experiment control) and must not be used otherwise. See section 9.3.2 for pin assignments. It is recommended to start instrument setup without anything connected to the control port.

4.6. Installation Troubleshooting

After completion of the software setup and connecting the HydraHarp 500 the device should be listed in the Windows Device Manager. Right click on the windows start button and select *Device Manager* to check if the device is free of conflicts and / or if the device driver is installed correctly. Under *PicoQuant TCSPC Devices* look for a device named *HydraHarp 500* and inspect its *Properties*.

A common source of problems is the shutdown behavior of Windows 10 and 11. It does not fully shut down by default but goes only into a state similar to hibernation in order to re-start more quickly. When new hardware is installed this can cause problems. If you missed this during hardware installation, do it at least when problems arise. In order to fully shut down Windows hold the shift key while clicking the shutdown button or run the command `shutdown /s /t 0` from the command prompt. It is also possible to permanently configure Windows for proper shutdown via the “power options” dialog.

If the HydraHarp 500 device driver is not installed or needs updating you can install it manually. When prompted to search for the driver, direct the driver wizard to search the location where you unpacked the setup files. Note that Windows 7 must have the most recent Windows updates applied in order to install the driver successfully.

If things are not working as expected you can also use the Windows system information facilities (*Start* > *Run* > *msinfo32).* In the System Information utility inspect *Software environment* > *System Drivers* to check if the HydraHarp 500 device driver PQCYUSB.SYS is correctly installed.

You can also repeat the software installation if necessary. To do so, first uninstall the software and repeat the setup procedure. Make sure the software is not installed in multiple places. If this does not resolve the problem, try a different computer. If problems persist, see section 8 for support.

4.7. Uninstalling the Software

Before uninstalling the HydraHarp 500 software you should back up all data files you might have created in the installation directory.

Do not manually delete any program files from the installation folder as this will prevent a clean uninstall process.

To uninstall the HydraHarp 500 software from your PC you may need administrator rights (depending on Windows version and security settings). From the list of installed applications select *PicoQuant – HydraHarp 500 vx.x* for un–installation. This will remove all files that were installed by the HydraHarp 500 setup program but not the user data that may have been stored. If there was user data in the program folders or any subfolders, these will not be deleted by the uninstall program. If intended you need to delete these files or folders manually. Nevertheless, it is recommended to back up valuable measurement data before uninstalling the software.

Note that uninstallation of the data acquisition software does not uninstall the device driver. This is because other software may still need it. You can delete the driver software from within Device Manager.

Note also that un–installation of the data acquisition software does not automatically uninstall the PQ File Info shell extension. It can be un-installed as a separate item through the standard Windows Control Panel mechanisms.

5. Software Overview

5.1. Starting the HydraHarp 500 Software

After correct installation the Windows Start Menu contains a shortcut to the HydraHarp 500 software. To start the software select *PicoQuant – HydraHarp 500 vX.X – HydraHarp 500 500*. Note that after switching the device on, you need to allow a warm–up period of about 20 minutes before using the instrument for serious measurements. You can use this time for set–up and preliminary measurements.

If the software cannot find a HydraHarp 500 device (or if there are driver problems) it will display a notification message, but it will still start. However, device dependent toolbar buttons and functions of the program will then be disabled. This allows you to use the software without hardware, e.g., to view or print files on another computer.

It is possible to use up to eight HydraHarp 500 devices on one PC. If multiple devices are installed then the first instance of the software will connect to the first device, the second to the second device and so on. Each of the software instances displays the serial number of the device it uses. An instance that does not find an unused device will open as a file viewer only. The HydraHarp 500 software keeps the settings in the registry separate for each device, based on the serial numbers.

Note, that the instances of the HydraHarp 500 software will be running completely independent of each other. If your application requires some kind of joint action of multiple HydraHarp 500 devices then you must design your own software based on the HydraHarp 500 programming library HH500Lib (see separate manual). If your objective is to combine multiple devices to obtain more detector channels you may need to synchronize the clocks of the devices. You also need to consider that it is not possible to prevent Windows from introducing unpredictable delays in communication with the hardware. The latter makes it impossible to, e.g., start measurements on multiple devices at the exact same time. See the HH500Lib manual and demos for partial solutions.

If the HydraHarp 500 is correctly installed and there are still hardware related errors, you can use the Windows device manager for troubleshooting (see the corresponding section above). If problems cannot be resolved, see section 8 for support. If possible, try the using the device with another computer.

For regular use of the HydraHarp 500 software you may want to create a shortcut for it on your Windows desktop. You can do this conveniently via a right mouse click on the start menu entry (*send to > Desktop – create shortcut*). Alternatively use the Windows Explorer to locate the file `HydraHarp 500 500.exe` in the directory you selected for installation and drag the icon onto the desktop. This will create a a shortcut to the executable file. If you have multiple devices connected and wish to launch the HydraHarp 500 software so that it uses a specific device then you can create a shortcut and edit it to pass the serial number as in the following example:

```
HydraHarp500.exe /serial=1001234
```

You can also start up the HydraHarp 500 software directly from a HydraHarp 500 data file by double–clicking on the file or dragging it onto the HydraHarp 500 shortcut. Note that this may be spoiled if you have other PicoQuant TCSPC devices installed on the same computer. The file name extensions may be assigned to other software.

5.2. The Main Window

The HydraHarp 500 software provides a measurement control interface to the HydraHarp 500 hardware and an online histogram display. Most prominently the main window accommodates the histogram display area.

Above the display area is the toolbar. Here you can access frequently used commands by simple mouse clicks. Next to the toolbar buttons you see the serial number of the device in use. Above the toolbar is the menu bar with additional commands. At the bottom of the histogram display area is a set of 'panel meters' showing count rates, count sums, and histogram peak characteristics. These will be updated continuously, some only when a measurement is in progress. Note the selector at the right of the panel meters. This selects the detector channel the meters are displaying. Instead of a single channel you can also select Sum, which then displays the summed rates from all detector channels. The panel meters can be enlarged by double-clicking them, which is useful when performing optical alignment or similar tasks when the PC monitor is some distance away.

In the top center of the display area a title line is shown. This can be double-clicked to edit the title. When editing the title, note that only the first line will appear in the display. The remaining lines are meant to be used as a file comment. All lines will be stored with the file data, with a maximum of 256 characters.

The main window is resizeable and the actual histogram display will adapt its size accordingly. If you make the window smaller than the minimum histogram display, two scrollbars will permit access to hidden window areas. Note that the actual size of the main window depends on the system font selected. It will scale to the size of the current system font. Screen resolutions below 800x600 are not suitable for serious work with the software.

Note that the position and size of the main window on the screen will be stored in the Windows registry and retrieved upon the next program start. The registry settings are kept separately for each user and device, provided he / she is logged on with a personal user account. Consult section 7.1 for further main window command descriptions. Toolbar, Menus, panel meters etc. will be explained in the next sections.

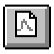
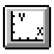
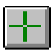
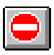
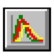
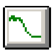
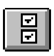
At the very bottom of the main window there is a status bar. The leftmost area of the status bar describes actions of menu items as you navigate through menus. Similarly it shows messages that describe the actions of toolbar buttons. The second status bar area from the left shows the current measurement status of the HydraHarp 500 . The rightmost area of the status bar indicates if the `<Caps>` and `<NumLock>` keys are latched down.

When the HydraHarp 500 software is running with functional hardware it continuously collects information about the input signals and the current acquisition settings. If these settings together with the input rates indicate possible errors, the software will display a warning icon in the status bar. The warning icon can be clicked to review the list of current warnings together with a brief explanation (see also section 9.1).

5.3. The Toolbar

The toolbar is displayed across the top of the HydraHarp 500 main window, below the menu bar. The toolbar provides quick mouse access to frequently used commands and tools. Note that some buttons may be grayed out (disabled) depending on the installed software components and / or the state of the hardware.

To hide or display the toolbar, choose Toolbar from the View menu (`<Alt>+V T`). The following table explains the individual buttons.

Click…	**to…**
	open a blank histogram with default control panel settings
	open an existing histogram file. Displays the *Open* dialog box, in which you can locate and open the desired file.
	save the current histogram data with its current name. If you have not named the file, the *Save As…* dialog box is displayed.
	copy the currently displayed curves to the clipboard (ASCII export).
	print the currently displayed histogram curves.
	display the *About…* window. This is where you can determine the version of your HydraHarp 500 software and hardware. Also provides links for updates etc.
	activate context sensitive help.
	launch the axis panel.
	launch the data cursor dialog.
	start measurement based on current HydraHarp 500 control panel settings
	stop measurement and histogram accumulation.
	launch the HydraHarp 500 control panel.
	launch the trace mapping dialog.
	launch the TTTR mode dialog.
	launch the filtered TTTR mode dialog.
	Launch the TTTR mode real–time correlator dialog.
	launch the dialog for Sequence Mode (SEQ).
	launch the general settings dialog.
	Launch the White Rabbit dialog

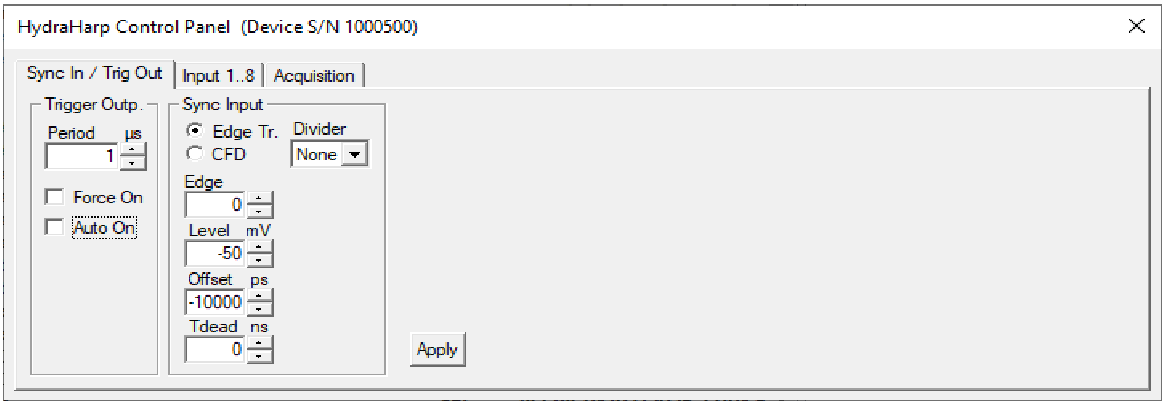
5.4. The Control Panel

The HydraHarp 500 control panel is a dialog box for setting the parameters for hardware adjustment and data acquisition. It is implemented as a 'non–modal' dialog box, i.e. it does not have to be closed before the main window can continue to operate. This way you can make changes to your settings in the control panel and watch their effect on a running measurement in the main window immediately. Nevertheless, you may close the control panel and restore it at any time by clicking the control panel button on the toolbar or pressing `<Alt>+C`.

The control panel consists of several pages (tabbed sheets) containing groups of edit boxes and other controls for related parameters. These pages and their respective controls are:

Sync–Input and Trigger Output

Trigger mode	radio buttons Edge Tr. / CFD
when above trigger mode Edge Tr. is selected:	
Trigger Edge	edit box and spin control
Trigger Level	edit box and spin control
when above trigger mode CFD is selected:	
CFD ZeroCr.	edit box and spin control
CFD Level	edit box and spin control
Offset	edit box and spin control
Tdead	edit box and spin control
Sync Divider	edit box and spin control
Trigger Outp. Period	edit box and spin control
Trigger Op. Force On	tick box
Trigger Op. Auto On	tick box

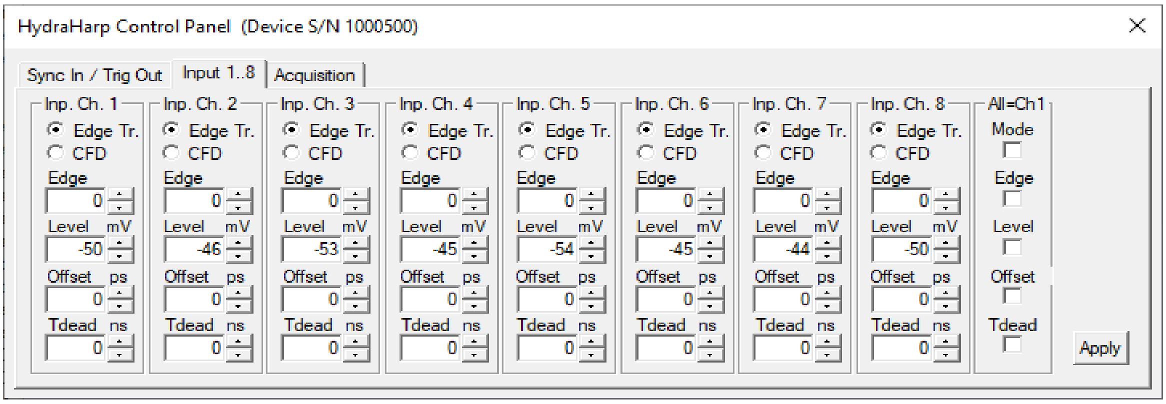


Input 1..8

Trigger mode	radio buttons Edge Tr. / CFD
when above trigger mode Edge Tr. is selected:	
Trigger Edge	edit box and spin control
Trigger Level	edit box and spin control
when above trigger mode CFD is selected:	
CFD ZeroCr.	edit box and spin control
CFD Level	edit box and spin control
Offset	edit box and spin control
Tdead	edit box and spin control

All=Ch1, Mode	tick box
All=Ch1, Edge/ZeroCr.	tick box, available only when all inputs ar set for the same trigger mode
All=Ch1, Level	tick box, available only when all inputs ar set for the same trigger mode
All=Ch1, Offset	tick box
All=Ch1, Tdead	tick box

Controls for channels that do not exist in the HydraHarp 500 being used will remain disabled (gray).

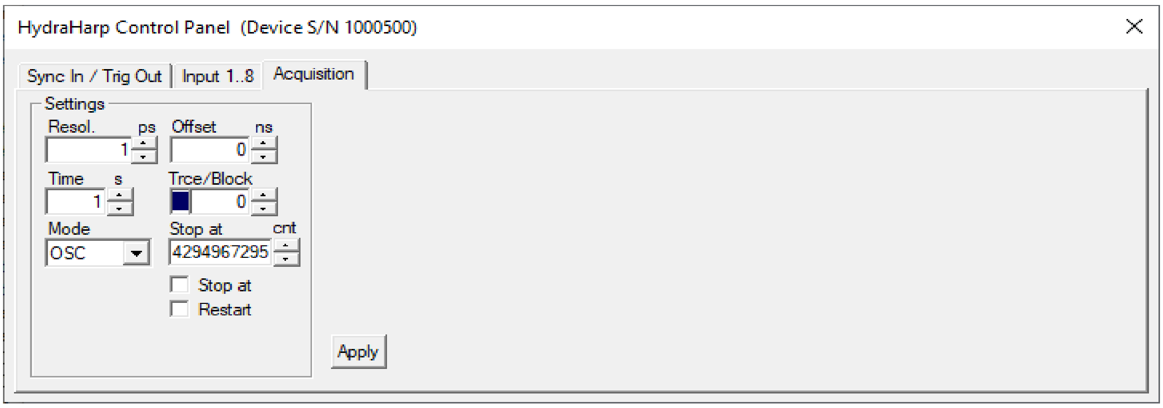


Input 9..16, 17..24, and so on

These tab pages only appear when the device has more than 8 input channels.
They have the same controls as the page for inputs 1..8 and follow the same logic of operation.

Acquisition

Resolution	edit box and spin control
Time	edit box and spin control
Offset	edit box and spin control
Trc/Block	trace color indicator, edit box and spin control
Restart	check box
Stop on Overflow	check box
Mode	drop-down selection box
Stop At	edit box and spin control



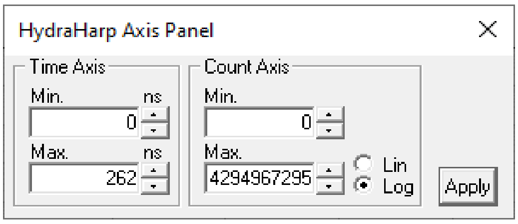
Edit boxes are for keyboard entry. The values must be confirmed with either the `<Enter>` key or the *Apply* button. The spin controls can be used to increment or decrement the value in the edit box. In this case the changes take effect immediately without need for hitting `<Enter>` or clicking *Apply*. Check boxes have their denoted effect when the tick is shown. They can be toggled with a mouse click. Groups of radio buttons are like check boxes but mutually exclusive.

Note that the settings of the control panel as well as the positions of control panel and main window on the screen will be stored in the Windows registry and retrieved during the next program start. The registry settings are stored separately per user. When a HydraHarp 500 data file is loaded, the control panel settings will change to reflect the settings stored in that file.

The individual control panel items are explained further in section 7.4 in the Controls and Commands Reference.

5.5. The Axis Panel

The axis settings panel is a dialog box for setting the axis range for the histogram display in the main window. It is implemented as a ‘non–modal’ dialog box, i.e. it does not have to be closed before the main window can continue its operation. This way you can make changes in the axis panel and watch their effect in the main window immediately. Nevertheless, you may close the axis panel and restore it at any time by clicking the axis panel button on the toolbar or pressing `<Alt>+A`. The panel will also open if you double–click the axes in the main window.



The axis panel consists of two groups containing edit boxes and other controls for related parameters. These groups and their respective controls are:

Time Axis

Minimum	edit box and spin control
Maximum	edit box and spin control

Count Axis

Minimum	edit box and spin control
Maximum	edit box and spin control
Linear	radio button
Logarithmic	radio button

The edit boxes are for keyboard entry. The values must be confirmed with either the `<Enter>`–key or the *Apply*–button. The spin controls can be used to increment or decrement the values in the edit box. In this case the changes take effect immediately without need for hitting `<Enter>` or clicking *Apply.* The mouse wheel can be used for fast spins as long as the cursor is in the corresponding edit box. Check boxes have their denoted effect when the tick is shown. They can be toggled with a mouse click. Groups of radio buttons are like check boxes but mutually exclusive.

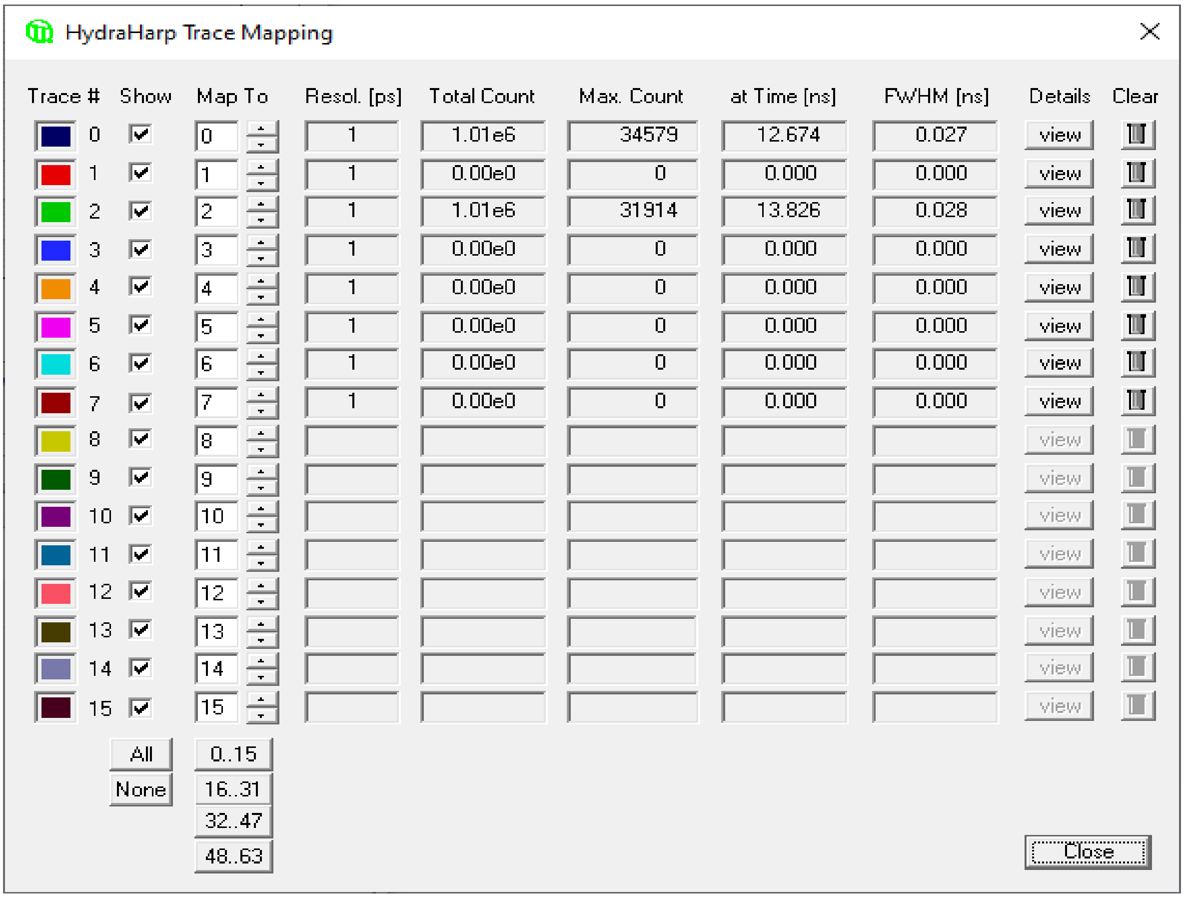
Note that the settings of the axis panel as well as the positions of the panel on the screen will be stored in the Windows registry and retrieved during the next program start. The registry settings are stored on a per user basis.

The individual axis panel items are discussed in section 7.5. “Axis Panel“ in the Controls and Commands Reference.

Hint: to begin, use logarithmic display to make sure even weak signals can be seen.

5.6. The Trace Mapping Dialog

The HydraHarp 500 software can record and store histograms in up to 512 data blocks in memory and files. Out of these, up to 16 traces can be displayed at the same time. It may seem odd that when 512 traces can be recorded that only 16 can be displayed. However, it is simply getting too cluttered when so many traces are drawn on the same screen. The limitation to 16 is still allowing reasonably distinguishable colors and keeping the view reasonably tidy. The *Trace Mapping* dialog is used to select the histograms to display. It also allows to view curve details and to delete unwanted blocks. You can use the *Trace Map* button on the Toolbar or click the curve color indicator in the control panel to launch the dialog.



In the *Trace Mapping* dialog you can tick the individual boxes '*Show*' to display a trace. You can also select the index of the memory block you wish to map the individual display traces to.

The dialog also provides some statistics on each trace (central matrix of figures). *Resolution* is the bin width of the histogram in picoseconds. Next is the *Max. Count*, the count in the highest point of the histogram. The column *at Time* shows the time corresponding to the *Max. Count* bin. Leftmost there is the Full Width Half Maximum (*FWHM*) of the histogram peak (usually meaningful only for IRF traces).

There are also some buttons for frequently required actions: The button '*view*' can be clicked to see more detailed information such as time of recording, acquisition settings and count rates. The button *All* can be clicked to tick all traces as shown, the button *None* does the opposite. The button *0..15* can be clicked to set the default mapping of trace 0..15 to block 0..15. Similarly, the buttons below allow mapping to the subsequent groups of 16 blocks. This allows quick access when there are more than 16 traces. The *Clear* buttons (trash cans) can be clicked to delete the contents of individual blocks.

Note that the Trace Mapping dialog is non–modal. This means the dialog can remain open while a measurement is in progress. Note also that a measurement can be running in a block that is not mapped or shown.

The individual trace mapping panel items are explained further in the section 7.6. "Trace Mapping" in the Controls and Commands Reference.

5.7. Other Dialogs

In order to keep this manual brief and readable, the dialogs described here in the overview chapter are limited to the most important ones the reader should know about before starting practical work. Additional dialogs will be described implicitly in the following sections in the context of specific measurement tasks. For information on all other dialogs the user is kindly referred to the controls and commands reference (section 7) or consult the on–line help facility of the software. Pressing F1 in an active dialog will open a corresponding help page.

6. Specific Measurement Tasks

6.1. Setting Up the Input Channels

This section provides help and instructions for the first basic steps of setting up the instrument. However, if you are running TCSPC measurements for the first time we strongly recommend you read the primer on TCSPC in section 3 first. Also consider the literature listed there.

In order to acquire any data, the input channels and the sync input of the HydraHarp 500 must be set to match their electrical input signals. All inputs have a programmable edge trigger (ETR) allowing the selection of the trigger edge (rising=1, falling=0) and the trigger level in mV. Dependent on the hardware model, the HydraHarp 500 may also have inputs that offer a Constant Fraction Discriminator (CFD) alongside the ETR. These inputs can then be configured to use either the ETR or the CFD. For specification details see section 9.3.1 and take note of the maximum ratings. For simplicity in the process of getting started we initially assume here that the ETR mode is used.

In case of coincidence correlation experiments using two or more detectors, the input channels are typically used for one detector each and data is collected in T2 mode. In that case the sync input can (but need not) be used for a detector as well. If a quick visualization of coincidence counts is required, it is also possible to use histogramming mode. In that case, one detector is connected to the sync input. Coincidence histograms will be collected for each input channel with respect to the sync input. In time–resolved fluorescence measurements with a pulsed excitation source (typically a laser) the sync input receives a sync signal from the laser. Here we focus on the latter, more common case. Perform the following steps to set up the input discriminators.

Using the Panel Meters

At the bottom of the main window you find a set of panel meters. These are very important during set–up. The meters showing units of cps (counts per second) in their title are rate meters. The leftmost rate meter shows the sync input rate. The next meter shows the channel input rate. Note the selector at the far right of the panel meters. This selects the channel the rate meters (except sync) are referring to. Select the channel you are currently setting up. Note that the rate meters use a fixed gate time of 100 ms. Their accuracy at low rates is therefore limited. They really only serve as a quick means of diagnostics and should not be used to obtain definitive measurement results.

The other meters show the histogramming rate, the total count in the histogram, the maximum (peak) count and the position of the maximum. All of these values depend on the input selector.

Note that the Input selector also has a selection option labeled *Sum*. In this case the meters are fed from the sum of all input channels. This is the default.

Setting up the Sync Input

For typical fluorescence decay measurements, the HydraHarp 500 needs an electrical sync signal from the light source. The PDL Series of diode lasers from PicoQuant provides this signal directly. If the laser does not provide an appropriate electrical sync signal (e.g.,some Ti:Sa lasers), a sync detector (photo diode) such as the TDA 200 must be used. The sync signal must consist of pulses with steepness and amplitude matching the specifications of the HydraHarp 500 and the trigger edge must be set to the leading edge (rising=1, falling=0). For example, a NIM type signal is appropriate. This is a steep negative pulse (0.5 to 10 ns wide, active edge falling) of typically −800 mV into 50 Ω. The HydraHarp 500 can actually handle ±1.2 V but large amplitudes may cause excess interference and cross-talk. between the inputs. Amplitudes around 100 to 200 mV (on all inputs) are best in terms of timing precision and lowest histogram ripple due to cross-talk. It may therefore be advantageous to attenuate NIM pulses by 10 or 15 dB. Lowest cross-talk. is typically achieved by using signals of similar amplitude on all inputs. SMA in-line attenuators of suitable bandwidth can be used to adjust this. The trigger level is adjustable for optimum timing precision. Initially you should set it to half of the sync pulse amplitude. Later it can be fine-tuned empirically.

Unless you are sure what kind of signal your sync source delivers, use a fast oscilloscope (50 Ω input) to check the pulse shape, polarity and amplitude. The leading edge should be steep (ideally 2 ns rise/fall time or faster), there should also be no excessive ringing. The pulse width should be at least 0.4 ns, the upper limit is not critical.

If the signal is satisfactory, connect the source to the sync input and start the HydraHarp 500 software if it is not yet running. A detector signal (CH1, CH2, ...) is not required at this point but it does no harm if it is also connected. Open the HydraHarp 500 control panel. Leftmost, there is a group of controls for the sync input. Find the edit box and spin control for the trigger level. The level should be set to a value around half of the amplitude of the sync pulses. Next, set the appropriate signal edge (typically the leading edge of your signal). The code 0 means falling, 1 means rising. Then look at the sync divider. It must be set so that the sync rate divided by the shown divider value remains under 85 MHz. For a slow source such as a flash lamp it must be set to None. Tdead is a programmable dead time for suppression of afterpulsing artefacts of some detectors. On the sync input it is rarely needed. Leave it at 0 for now.

If the sync source is active, the sync rate will now be displayed at the bottom left rate meter in the HydraHarp 500 main window. Note that the sync rate meter internally corrects for the chosen divider setting, so that the meter always shows the undivided input rate. The meter display should therefore match the rate delivered from the source. Note that meaningful use of the divider requires periodic sync signals. The rate meter will be refreshed every 0.1 to 1 seconds, as determined by the value in the General Settings Dialog. The sync rate should be displayed very accurately unless it is very low. Large fluctuations or occasional zeros indicate an incorrect discriminator level setting or an unstable sync signal. Try varying the discriminator level to obtain a stable sync rate display. If the rate is stable and at the expected value, you can proceed to setting up the other inputs. A last fine tuning of the trigger level can be done when the detector inputs are up and running. Note that it may be impossible to get a stable sync rate reading at very low sync rates. This is because the rate counter uses a prescaler and a gate time of 100 ms and if the signal period is too low the readings will fluctuate. You will then have to set the level based on what you know about the pulse amplitude and verify it in an actual measurement.

Setting up the Photon Detector Inputs (Channel 1…N)

Depending on the hardware model and purchase options your HydraHarp 500 may have up to 16 input channels. The input signals must consist of pulses with steepness and amplitude matching the specifications and the trigger edge should be set to the leading edge (rising=1, falling=0). For example, a NIM type signal is appropriate. This is a steep negative pulse (0.5 to10 ns wide, active edge falling) of typically −800 mV into 50 Ω. The HydraHarp 500 can actually handle ±1.2 V but large amplitudes may cause excess interference and cross-talk. between the inputs. Amplitudes around 100 to 200 mV (on all inputs) are best in terms of timing precision and lowest histogram ripple. It may therefore be advantageous to attenuate NIM pulses by 10 or 15 dB. Lowest cross-talk. is typically achieved by using signals of similar amplitude on all inputs. SMA in-line attenuators of suitable bandwidth can be used to adjust this. Note that popular TTL–SPAD–detectors (e.g., Perkin–Elmer/Excelitas SPCM–AQR) may deliver positive pulses of ~3 V and must be connected through an attenuator or a pulse inverter with attenuation (PicoQuant SIA 400). **Connecting TTL signals directly will cause damage to the HydraHarp 500!**

PMTs should be connected through a preamplifier (10 to 20 dB). MCP–PMT detectors should be connected through an amplifier with slightly higher gain. All accessories are available from PicoQuant. Be sure to switch the high voltage supply of PMTs off and allow their electrodes to discharge before connecting / disconnecting them. Their high voltage charge may damage the preamplifier. Observe the allowed input signal levels including those of the pre–amplifier. Again, in a new experimental setup, to be absolutely sure, please check your detector pulses as well as the preamp output with a fast oscilloscope. Start timing is on the leading edge, so it should be steep. Ringing and overshoot should be as small as possible. Do not over–illuminate the detector to avoid damaging it.

If the signals are appropriate, connect a detector at channel 1. Starting with only one detector makes the first steps easier. Preliminary adjustments can actually be done with uncorrelated light, e.g. daylight. To protect your detector, use strongly attenuated light, even when the detector is off. Start the HydraHarp 500 software and open the Control Panel. Look at the Control Panel section for the chosen channel. Initially set the trigger level to half the expected pulse height and set the trigger edge (rising=1, falling=0) to match the leading edge of your detector pulses. Tdead is a programmable dead time for suppression of afterpulsing artefacts of some detectors. Leave it at 0 for now.

Using the input selector at the far right of the rate meters, select the channel you are currently using. The rate meter next to the sync rate meter will now show the input rate at that channel. Make sure that there is actually a signal coming in (some strongly attenuated light on the detector) and try to adjust the trigger level. To monitor this, watch the count rate meter. Precise tuning of the level settings is only critical for PMTs or HPDs. For such detectors an initial setting of the trigger level of -50 mV (assuming a non-inverting amplifier, otherwise +50mV) should be approximately right to suppress electrical noise. Moving this level towards zero should result in an increased count rate (probably mostly noise). If you move the trigger level farther away from zero, you will see a reduced rate that will incur losses of true counts if you go too far. Ideally you should stay just slightly above the noise level. If you still see millions of counts per second at -100 mV, reduce the detector illumination. Your detector may be at risk at such high count rates. You can then also try out responsivity to illumination changes. For SPAD detectors that typically deliver pre–shaped pulses of constant amplitude the setting of the trigger level is very simple. Just set the trigger level to approximately half of the pulse amplitude.

To actually collect histograms, select a measurement range large enough (determined by the chosen resolution) and a display range to cover your sync interval (i.e. $1 / f_{sync}$) if possible. Set Offset = 0 and StopAt = 4,294,967,295. Start a measurement in oscilloscope mode with e.g., 1 second acquisition time (see the next section for instructions). Once counts are coming in, you can try to set an upper limit for the count axis so that the histogram is scaled for best viewing. Start with logarithmic display so that weak signals can be seen.

If you used uncorrelated light so far, you may now want to move on to time–correlated measurements. With a detector signal that is e.g., induced by a laser or a 'fake' electrically derived from the laser sync, you can try to obtain a histogram that should be a narrow peak, as opposed to the flat distribution of uncorrelated light. This requires an experimental setup that delivers the signals at the two input channels with a more or less constant relative delay. In histogramming mode this delay must be such that the Channel 1..N (detector) signal comes later than the sync signal (forward start–stop mode). In addition, the delay must be chosen so that it fits in the measurement range of the histogrammer. To obtain such a timing it may be necessary to adjust the input offsets or the relative cable lengths, while also considering any optical delays.

If the rough setup is complete and a time–correlated signal peak is present in the histogram you may want to switch to a higher resolution. For optimization purposes you can then try to slightly vary the trigger levels (including sync) for best timing response. The Full Width at Half Maximum (FWHM) of the signal peak is displayed on-line as a figure of merit for instrument response. If there is no clear optimum for any level, return to the center of its stable working range. You can then also adjust the input offsets to place the signal properly within the time axis boundaries of the sync frame. You can also adjust your axis limits to optimize the display. Note that you can use the mouse wheel for quick changes of any spin control. Note also that you can double–click all the rate meters to enlarge them as separate windows. This is useful for optical alignment work if the PC screen is some distance away from your optical table.

Input Troubleshooting

Whenever there is a problem, first check your cabling, detector power supply, and sync source. To be absolutely sure, check the signals with a fast oscilloscope (50 Ω input!). **Never use TTL signals directly**. Never try to deliver the signals to multiple 50 Ω loads in parallel, using simple T–pads, use proper power splitters instead. If you have multiple detectors, try to test your setup with one detector first to keep things simple. Never forget to let detectors using high voltage discharge before connecting/disconnecting.

If you cannot get a stable sync rate reading with expected values, there may be several reasons:

- there is no proper sync signal (voltage, polarity, pulse width, frequency)
- the sync trigger level setting is inappropriate
- the sync divider setting is inappropriate
- Note that at small rates the meter display may fluctuate between some discrete values.

Reasons for zero input channel counts are:

- wrong input selection for the rate meters
- no or inappropriate signal (voltage, polarity, pulse width, frequency)
- inappropriate trigger levels
- preamplifier or detector failure

Note that all meters are updated at the display refresh rate you have selected in the general settings dialog. In oscilloscope mode the update frequency is equal to the acquisition time. If nothing seems to happen at all, you may be in oscilloscope mode with a very long acquisition time set.

Once count rates > 0 are being displayed you should then also see counts appearing in the histogramming rate meter, if your measurement range is large enough. Make sure the offset value in the Acquisition tab is set to 0. If there are histogramming counts but no histogram is building up on the screen, check the time and count axis bounds. It is best to start with a wide display range and then narrow it down. Similarly, a logarithmic count axis setting is the safest way to see even small histograms. If your measurements stop earlier than expected, make sure the Stop At level is not set to less than 4,294,967,295 unless you have experimental reason to limit the counts.

During the set-up process you should pay attention to the warning icon that may appear at the bottom right of the main window. When the HydraHarp 500 software is running with functional hardware it continuously collects information about the input signals and the current acquisition settings. If these settings in combination with the input rates indicate possible errors, the software will activate the warning icon.

The warning icon can be clicked to display a list of current warnings together with a brief explanation of each warning (see also section 9.1). Note that the software can detect only a subset of possible error conditions. It is therefore not safe to assume “all is right” just by seeing no warning. On the other hand, if any of the warnings turns out to be an unnecessary nuisance, e.g., because your specific measurement conditions will expectedly cause it, you can disable that warning via the general settings dialog (see section 7.7).

In ETR mode of the onputs is possible to set a hysteresis of the input comparators larger than the default value. This setting can be made though the general settings dialog (see section 7.7). It applies to all inputs simultaneously, including the sync input. There are only two choices: the default value of about 3 mV and the large hysteresis value of about 35 mV. The larger hysteresis may in some cases help to suppress noise artefacts on the input signals. Consider this only a last rescue when it is impossible to eliminate the noise at its origin.

Using the Timimg Inputs in CFD mode

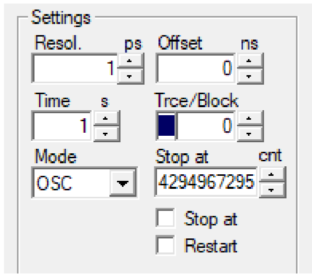
Dependent on the hardware model, the HydraHarp 500 may also have inputs that offer a Constant Fraction Discriminator (CFD) in addition to the ETR. These inputs can then be configured to use the CFD instead of the ETR. The CFD is useful when the amplitude of the detector pulses fluctuates, as observed in PMT, MCP and Hybrid detectors. Timing uncertainty can then be reduced using the CFD. Similarly it may help with SNSPDs when the effective pulse height fluctuates due to incomplete return to baseline upon preceeding pulses. Note that the timing benefit of the CFD goes along with a prolonged dead time. Please see section 3.4 for more information on how a CFD works. For specification details of the HydraHarp 500 ’s CFD see section 9.3.1 and take note of the maximum ratings. One important detail to note here is that the HydraHarp 500 ’s ETR can operate on positive and negative going signals while the CFD can only be used with negative going signals. This is compatible with almost all typical detectors requiring a CFD. If your detector is an exception in this regard it may still be possible to make it compatible by means of an inverting preamplifier.

Whether or not a given input channel has the CFD feature can be seen in the control panel. If the corresponding radio button is grayed out then the CFD is not available. Otherwise, clicking that radio button will switch the input to CFD mode. The configuration field “Edge” below will then change its label to “ZeroCr”. When switching from ETR to CFD the Level (provided it is nagative) can typically be kept at the same level that worked well in ETR mode. However, in order to improve timing precision it can be experimantally varied up or down. At the same time it can be optimized to suppress (discriminate) small detector pulses that are likely originating from noise. The CFDs Zero Cross setting can similarly be varied to optimize/compromise between the suppression of input noise and timing precision.

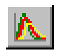
6.2. Setting Up and Running Interactive Measurements

The primary mode of operation of the HydraHarp 500 software is interactive histogramming. This is what the main window of the software is dedicated to. The user can set up measurement parameters, start measurements and immediately see histogram data on the screen. In further sections, e.g., on TTTR mode (time tagging), you will learn about other modes of operation with less user interaction that will collect data straight to disk without much immediate visualization. Here, we focus on the interactive histogramming mode of operation.

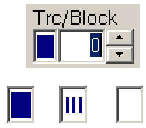
To set up measurement parameters use the HydraHarp 500 control panel. The control panel can be opened by clicking the control panel button on the toolbar or by pressing <Alt>+C.



In the control panel section 'Acquisition - Settings' you can set the resolution (time per bin), the offset, the measurement time, and the block of memory to use for this measurement. To begin, use a measurement time of 1 second and an offset of 0. There are always 65,536 time bins per histogram. Histograms can be recorded and stored in 512 memory blocks. Out of these up to 16 traces can be displayed and one ‘active’ block can be used for a measurement. You can designate the active memory block you wish to use for the next measurement by selecting the block number in the control panel. Never forget to select a new block when collecting new data and old data is intended to be preserved.



Use the trace mapping dialog to select up to 16 traces for display. Make sure the memory block you measure into is mapped to a display trace that is switched on, so that you can see the trace. You can reach the trace mapping dialog from the toolbar or by clicking on the trace color indicator next to the block selector.



The color indicator shows the trace color that the chosen block is currently mapped to. If it is a solid square, the curve is mapped and shown. If it is mapped but not shown, the indicator shows a small striped square. If the curve is not even mapped for display the indicator remains white.

There are two basic histogramming modes for interactive measurements: Oscilloscope and Integration mode. Oscilloscope mode repeatedly collects histograms with a fixed measurement time and displays them on the screen. This lets you see fast changes in the histogram, e.g., for optical setup and adjustments. Usually this only makes sense with relatively strong signals and short acquisition times. Integration mode is usually operated with longer acquisition times. In this case the histogram continues to grow over a longer time and the display is updated at regular intervals, so that the accumulation process can be observed.

To start a measurement with the current control panel settings, use the start button (GO) on the toolbar or press <Alt>+G.

To stop a measurement use the stop button on the toolbar or press <Alt>+S.
Note that a measurement may automatically stop and / or restart, dependent on the current settings of ‘*stop on overflow*’ and ‘*restart*’ in the control panel.

Note: to actually run a meaningful measurement you will need to set up the input channels, most importantly with appropriate voltage levels as outlined in the previous section. Also allow a warming–up period of about 20 minutes (depending on lab temperature) before using the HydraHarp 500 for final measurements. You can use this time for set–up checks and preliminary measurements.

The rate meters (bottom of main window) permit visual control of the data acquisition. Note the selector at the far right of the rate meters. This selects the channel the rate meters are referring to. Select the channel you are currently using. The meters are often too small to view from further away, e.g., when adjusting the optical setup. You can in this case simply double–click the rate meter of interest. This opens a large meter display that you can then re-size and move on the screen as you like.

Once you have established standard settings for your experimental setup you may want to save them to a file. The control panel settings can then be recalled at any time by loading that file. The program stores all settings together with the histogram data of those curves in memory (max. 512) that have been filled by a measurement. In addition to this, all settings are stored in the Windows registry, so that at program startup you find the control panel as it was when you last closed the HydraHarp 500 software.

6.3. Time Tagged Mode Measurements

Time–Tagged Time–Resolved (TTTR) mode in its simplest form allows the recording of individual count events directly to hard disk without immediately forming histograms.

In classic TTTR mode, in addition to the start–stop timing with picosecond resolution, the timing of the events with respect to the beginning of the whole measurement is recorded in the event records. This is particularly interesting where the dynamics in a fluorescence process are to be investigated. The availability of the time–tags permits photon burst identification, which is of great value e.g., for Single Molecule Detection (SMD) in a liquid flow. Other typical applications are Fluorescence Correlation Spectroscopy (FCS) and Burst Integrated Fluorescence Lifetime (BIFL) measurements. Together with an appropriate scan controller, TTTR mode is also suitable for ultra fast Fluorescence Lifetime Imaging (FLIM). Another great application area of time tagging is quantum optics with coincidence counting and correlation.

The HydraHarp 500 actually supports two different Time–Tagging modes, T2 and T3 mode, which will be explained further below. When referring to both modes together we use the general term TTTR.

6.3.1. System Requirements

In cases where the Time–Tagging modes are to be used with high continuous count rates (say > 5 Mcps) the PC system must meet some special performance criteria. The reason for this is the relatively large amount of data being generated in TTTR mode. In order to prevent an overflow in the recording, the data must be transferred to the computer in real–time. This requires a modern PC with a fast I/O subsystem. A recent, at least dual core CPU running at 2 GHz or more is required. For the best possible performance in TTTR mode a modern solid state disk with high throughput is recommended. If it is intended to make use of the full TTTR throughput of a HydraHarp 500 (up to 90 Mcps via USB 3) then the hard disk must be able to handle sustained write rates of 360 MBytes/s. This can be achieved with RAID arrays or solid state disks. Network storage is usually too slow.

6.3.2. T2 Mode

In T2 mode all timing inputs of the HydraHarp 500 including the sync input are functionally identical. There is no dedication of the sync input channel to a sync signal from a laser. It may be left unconnected or can be used for an additional detector signal. In this case the sync divider must be set to “None”. Usually the regular inputs CH1..N are used to connect photon detectors. The events from all channels are recorded independently and treated equally. In each case an event record is generated that contains information about the channel it came from along with the arrival time of the event with respect to the overall measurement start. The timing is recorded with the highest resolution the hardware supports (1 ps). Each T2 mode event record consists of 32 bits. There are 6 bits for the channel number and 25 bits for the time–tag. If the time tag overflows, a special overflow record is inserted in the data stream, so that upon processing of the data stream a theoretically infinite time span can be recovered at full resolution. Dead times exist only within each channel but not across the channels. Therefore, cross correlations can be calculated down to zero lag time. This allows for powerful applications such as coincidence correlation and FCS with lag times from picoseconds to hours. Autocorrelations can also be calculated at the full resolution but of course only starting from lag times larger than the dead time.

The 32-bit event records need to be continuously streamed to the host PC, ideally without losses. Given the dead time of the TDCs of Td=680 ps it is theoretically possible that event records are generated at a maximum rate of 1/Td = 1,47 GHz. The total rate increases even more when multiple channels are used. It is obvious that such data rates cannot be transferred over USB continuously without losses. On the other hand, given typical photon statistics one can (and should) distinguish between peak rates and average rates. The latter are typically much smaller than the peak rates during bursts. The system is therefore designed to use FIFO (First In First Out) buffers that can temporarily hold a certain number of events in a burst so that the bus transfer must only deal with moderate average rates. In order to allow a high input burst rate the HydraHarp 500 uses in each channel a fast front-end FIFO buffer that can handle bursts of up to 2048 events at the maximum rate of 1/Td = 1,47 GHz. This is followed by a large but slower FIFO buffer for all channels, capable of holding up to 134,217,728 event records. This large secondary FIFO buffer ensures that no data is lost due to inevitable task switches and interruptions of the USB transfers on the host PC side. Even if the average read rate of the host PC is limited, bursts with much higher rate can be recorded for some time. Only if the average count rate exceeds the readout speed of the PC over a long period of time, a FIFO buffer overrun can occur. In case of a FIFO buffer overrun the measurement must be aborted because data integrity cannot be maintained. However, on a modern and well configured PC a sustained average count rate of up to 90 Mcps is possible. This total transfer rate must be shared by the inputs used. For all practically relevant photon detection applications the effective rate per channel is typically more than sufficient. If there are very intense bursts at the input of the front-end FIFO it may happen that events are lost. This is indicated to the software by means of a hardware flag so that the user can be informed of such losses. The user must then decide if the losses can be tolerated for the given experiment (see also section 9.1 on warnings).

For maximum throughput, T2 mode data streams are normally written directly to disk, without preview other than count rate and progress display. However, it is also possible to analyze incoming data ”on the fly”. The HydraHarp 500 software provides a real-time correlator for preview during a T2 mode measurement (see section 6.3.7).

The HydraHarp 500 software installation provides demo programs to show how T2 mode files can be read by custom software (see the folder `filedemo` under the chosen software installation folder). The implementation of custom measurement programs requires the HydraHarp 500 programming library, which is provided as a separate software package for download. A relatively advanced high-level API package for Python called “snAPI” is also available for free. It readily provides data collection and file writing methods as well as many real-time analysis methods such as intensity and coincidence time traces, FCS and $g^{(2)}$ correlation. Other alternatives for advanced T2 data collection and analysis are the new UniHarp software as well as the SymPhoTime and QuCoa software suites offered by PicoQuant. SymPhoTime is focused on life science microscopy applications while QuCoa is oriented towards typical quantum optics applications.

6.3.3. T3 Mode

In T3 mode the sync input is dedicated to a periodic sync signal, typically from a laser. As far as the experimental setup is concerned, this is similar to classic TCSPC histogramming. The main objective is to allow for high sync rates which could not be handled in T2 mode. Accommodating the high sync rates in T3 mode is achieved as follows: First, the sync divider is employed as in histogramming mode. This reduces the sync rate so that the channel dead time is no longer a problem. The remaining problem is now that even with the divider, the sync event rate may still be too high for collecting all individual sync events like in ordinary T2 mode. Considering that sync events are not of primary interest, the solution is to record them only if they arrive in the context of a photon event on any of the input channels. The event record is then composed of two timing figures: 1) the start–stop timing difference between the photon event and the last sync event, and 2) the arrival time of the event pair on the overall experiment time scale (the time tag). The latter is obtained by simply counting sync pulses. From the T3 mode event records it is therefore possible to precisely determine which sync period a photon event belongs to. Since the sync period is also known precisely, this furthermore allows reconstructing the arrival time of the photon with respect to the overall experiment time.

Each T3 mode event record consists of 32 bits. There are 6 bits for the channel number, 15 bits for the start–stop time and 10 bits for the sync counter. If the counter overflows, a special overflow record is inserted in the data stream, so that upon processing of the data stream a theoretically infinite time span can be recovered. The 15 bits for the start–stop time difference cover a time span of 32,768×R where R is the chosen resolution. At the best possible resolution of 1 ps this results in a span of 32.7 ns. If the time difference between a photon and the last sync event is larger, the photon event cannot be recorded. This is the same as in histogramming mode, where the number of bins is larger but also finite. However, by choosing a suitable sync rate and a compatible resolution R, it should be possible to reasonably accommodate all relevant experiment scenarios. R can be chosen in a wide range, starting with the device's base resolution and then continuing by repeated doubling of the time bin width.

Dead time in T3 mode is the same as in the other modes. Within each photon channel, autocorrelations can be calculated meaningfully only starting from lag times larger than the dead time. Across channels dead time does not affect the correlation so that meaningful results can be obtained at the chosen resolution, all the way down to zero lag time. This requires custom software.

The 32 bit event records are queued and forwarded to the host PC in the same staggered FIFO architecture as described in the section on T2 mode above. Accordingly, a sustained average count rate of up to 90 Mcps is possible in T3 mode too, while now the sync events do not consume any transfer bandwidth.

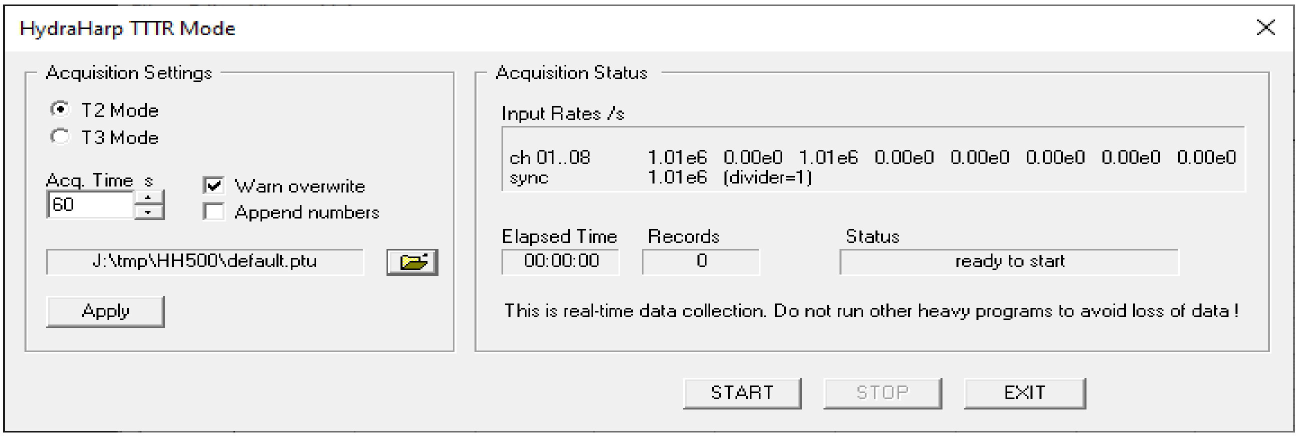
For maximum throughput, T3 mode data streams are normally written directly to disk. However, it is also possible to analyze incoming data ”on the fly”. One such analysis method is the on–line correlation implemented in the HydraHarp 500 software (see section 6.3.7). Other specialized analysis methods must be implemented via custom software. The HydraHarp 500 software installation provides demo programs showing how T3 mode files can be read (see the folder `filedemo` under the chosen software installation folder). The implementation of custom measurement programs requires the HydraHarp 500 programming library, which is provided for free download as a separate software package. A relatively advanced high-level API package for Python called “snAPI” is also available for free. It readily provides data collection and file writing methods as well as many real-time analysis methods such as intensity and coincidence time traces, FCS and $g^{(2)}$ correlation. Other alternatives for advanced data collection and analysis are the new UniHarp software as well as the SymPhoTime and QuCoa software suites offered by PicoQuant. SymPhoTime is focused on life science microscopy applications while QuCoa is oriented towards typical quantum optics applications.

6.3.4. Running a basic TTTR Mode Measurement

A TTTR mode measurement (T2 or T3 mode) will typically be started after all control panel settings have been tested in normal interactive histogramming mode (oscilloscope or integration). The acquisition time (measurement time) and the file for saving the data are the only parameters that can be set separately.

A typical approach to set up a TTTR mode measurement would be by first starting oscilloscope mode with an acquisition time of e.g., 1 second. Then all control panel settings should be optimized to reliably obtain the expected data.

Once all settings are satisfactory, click the “*TTTR Mode*“ button on the toolbar. This will bring up the “*TTTR Mode*“ dialog.



The dialog section *Acquisition Settings* allows selecting the measurement mode (T2 / T3), overall acquisition time, and file name. Note that switching between T2 and T3 mode takes some time because the hardware must be reconfigured. Normally such a switching should not occur often because the two modes usually require a different experiment setup.

The section *Acquisition Settings* also has two tick boxes for the handling of existing files. You can turn on a warning and / or automatically have numbers appended to the file names, so that you can conveniently perform series of measurements. The file name is shown in red if the file already exists. The button with the file icon will open a standard Windows file dialog. You can select an existing file or choose a new name. The HydraHarp 500 TTTR mode files have the extension "`.ptu`". For maximum count rate throughput you should choose a file destination on a fast local hard disk as outlined above. Network drives are often too slow.

The dialog section *Status* shows elapsed time, the count rates and the number of collected records. The number of shown input rates depends on how many channels your HydraHarp 500 has. Below these figures is a status line showing what is currently happening. Further below there are buttons for Start, Stop and Exit. Start and Stop control the actual TTTR measurement run. Exit is for leaving the TTTR mode dialog.

The TTTR mode measurement will start as soon as you click the Start button. You will then be able to watch the progress of your measurement in the status boxes showing the elapsed measurement time and the number of events that have been recorded. Note that this includes overflow and marker records. Therefore, the number of records will be somewhat larger than the true photon counts. The overflow and marker records will later be filtered out by the data analysis / processing software.

Next to the status line you may see a warning icon. When the TTTR mode dialog is running it continuously collects information about the input signals and the current acquisition settings. If these settings in conjunction with the input rates indicate possible errors, the software will display the warning icon. The warning icon can be clicked to show a list of current warnings together with a brief explanation of each warning (see also section 9.1). In order to fix wrong settings you will have to close the TTTR mode dialog and return to the control panel.

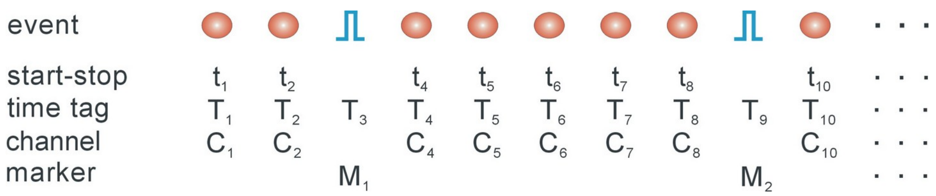
A measurement can be stopped at any time by clicking the Stop button. The data recorded up to this point will be stored in the file. When the measurement has completed, the Stop button will be grayed out (disabled). Use the Exit button to return to the normal interactive mode. Again, this will take some time for hardware reconfiguration.

As outlined above, TTTR data collection at high rates is a demanding real-time streaming process. The hardware and software must ensure not to lose any data. In order to implement this efficiently the HydraHarp 500 software employs multiple threads (concurrent CPU processes). A first thread continuously reads the HydraHarp 500 ‘s hardware FIFO and puts the retrieved data in a software queue. A second thread concurrently reads this software queue and writes the data to disk. If real-time correlation is being performed (see section 6.3.7) then this is done in further separate threads. User interface and interaction are handled in yet another thread. Multicore CPUs are particularly useful here as they can run the threads in parallel rather than switching between them. There are two typical error scenarios that you may encounter in this process. The first is a situation where the first thread does not empty the hardware FIFO quickly enough and the FIFO runs full. The software then reports the error FIFO_OVERRUN. In order to avoid this you may have to reduce the input data rate. Another error situation may result when the second thread cannot write to disk quickly enough and the software queue runs full. The software then reports the error STORAGE_QUEUE_OVERRUN. In order to avoid this you may want to check the write speed of your hard disk and see how it can be improved.

6.3.5. External Markers

Often it is desirable to synchronize TCSPC measurements with other events or processes of complex measurement tasks. In order to perform e.g., Fluorescence Lifetime Imaging, the spatial origin of the photons must be recorded as well as their timing. For this purpose one needs a mechanism to assign external synchronization information to the TCSPC data. In the special case of Fluorescence Lifetime Imaging, conventional systems used on–board memory and switch to new blocks of memory upon arrival of e.g., a pixel clock pulse. Accommodating the large amount of data generated by the 3–dimensional matrix of pixel co–ordinates and lifetime histogram bins is a serious challenge. Even with modern memory chips, this approach is limited in image size and / or number. In addition, it is expensive, and implies loss of information about the individual photon arrival times. To solve the problem in a much more elegant manner, the TTTR data stream generated by the HydraHarp 500 can contain markers for synchronization information derived from an imaging device, e.g., a scan controller. For this purpose the control port of the HydraHarp 500 provides four LVTTL inputs for synchronization signals M1..M4 (see section 9.3.2 for the connector specification).

The figure below illustrates how the external marker signals are recorded in the data stream.



Bullets denote a photon, blue pulses denote a marker signal. The external markers are treated almost as if they were regular photon event records. A special channel code allows to distinguish true photon records from marker records. Software reading the TTTR file can thereby filter out the markers e.g., for line and frame clock in imaging applications. This makes possible to reconstruct the 2D image from the stream of TTTR records, since the relevant XY position of a scanner can be determined during the data analysis. The data generated this way is nearly free of redundancy and can therefore be transferred in real–time. The image size is unlimited both in XY and in count depth. Since there are four such synchronization signals, all imaging applications can be implemented and even other experiment control signals can be recorded. This marker scheme is a very special feature of PicoQuant's TCSPC electronics. It may be worth noting that inventing this technology enabled PicoQuant to develop the trend-setting MicroTime 200 Fluorescence Lifetime Microscope.

The LVTTL compatible inputs accept the synchronization signals that will be recorded as markers. The active edges of these signals can be chosen in the general settings dialog (available through the Toolbar). Both high and low state must be at least 50 ns long. The period may therefore (in principle) be as short as 100 ns but data bus throughput constraints will apply. Each marker creates an additional TTTR record, so that one must ensure not to swamp the data stream with too many marker records. In case of data bus congestion markers take precedence over photon records, so that excessive marker traffic can suppress photon records. In fast imaging applications it is therefore recommended not to use a pixel clock but a line clock only. Since each photon has a time tag, it is usually not necessary to use an additional pixel clock. Instead, virtual pixels can be defined by subdividing the scan lines in time. The accuracy of marker timing is on the order of 50 ns. Note, however, that in T3 mode it can never be resolved better than one sync period.

A programmable marker hold-off time can be used to suppress glitches on the marker signals that some poorly designed scan hardware or cable reflections might create. The idea is that when a marker signal was detected the next (spurious) marker will be suppressed if it occurs within the hold-off time after the first detection. The hold-off time can be chosen in the software settings dialog available through the Toolbar.

6.3.6. Using TTTR Mode Data Files

For diagnostic purposes you may reload a T3 mode file into the HydraHarp 500 software. The limitation is that you will only be able to form a histogram over the start-stop times in your T3 mode data. The time-tag information will not be used here. The software will recognize that you are loading a T3 mode file and how many records are contained in it. It will then prompt you for a range to use for histogramming. The histogram will go to blocks 0..N-1 of the histogram memory where N is the number of channels that were used in the T3 mode measurement. A TTTR mode file also contains all control panel settings that were active at the time of the measurement run. After loading a TTTR mode file, you will find the document title reading “Histogram from...”. If you choose to save such data you will have to give it a new file name (`*.phu`). This is because now a histogram has been formed, and saving it with the same file name would destroy the original TTTR mode file. However, you may save the previously formed histogram as if it were obtained in normal interactive mode, to a standard HydraHarp 500 histogram data file (*.phu).

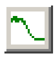
Reloading T3 mode files serves as a quick diagnostic tool only. For T2 mode files such a feature is currently not available. Further processing or analysis of TTTR mode data must therefore be performed through external data analysis software. Such software is available from PicoQuant for a wide range of analysis tasks (under the product names snAPI, UniHarp, SymPhoTime and QuCoa). Further specialized analysis can be performed by dedicated custom software. If you wish to save the cost for the commercial TTTR analysis software or if you require special analysis algorithms you may want to program your own analysis software. For development of your own custom programs, please refer to snAPI (Python) or our demo code in other programming languages for loading `.phu` and `.ptu` files. The latter is included in your HydraHarp 500 software download and will be installed by the software setup into the subfolder `filedemo`. Also see section 9.2 for the file format specifications. The paragraph below gives only an outline.

The first part of a TTTR mode file is a header with the basic setup information, similar to that of the other modes. What follows after the header is a sequence of 32 bit TTTR records. The TTTR records in the file consist of different pieces of information in groups of bits. These pieces of information must be extracted by a bit masking / shifting operation. Their specific layout is different for T2 and T3 mode. In both cases, in addition to extracting the bit fields, the software must step through the whole file and interpret the overflow records and correct the overall time axis accordingly. Details on how this can be done should be looked up in the demo programs that are installed as part of the software distribution.

6.3.7. TTTR Mode Measurements with Real–Time Correlation

An important application of the HydraHarp 500 and its TTTR mode is Fluorescence Correlation Spectroscopy (FCS). While TTTR data collection and off–line software correlation has been used for quite some time, PCs have meanwhile become fast enough to calculate the correlation function “on the fly” while the data is being collected. This capability can be very useful in setting up and monitoring FCS experiments. The HydraHarp 500 software provides such a real–time software correlator for T2 and T3 mode. Note that this is not a correlator for $g^{(2)}$ correlations common in quantum optics. If you need this type of correlator please consider PicoQuant’s QuCoa software.

Although it collects the same type of data as regular TTTR mode, the real–time correlator has its own button on the Toolbar.



The next figure shows the TTTR mode correlator dialog.

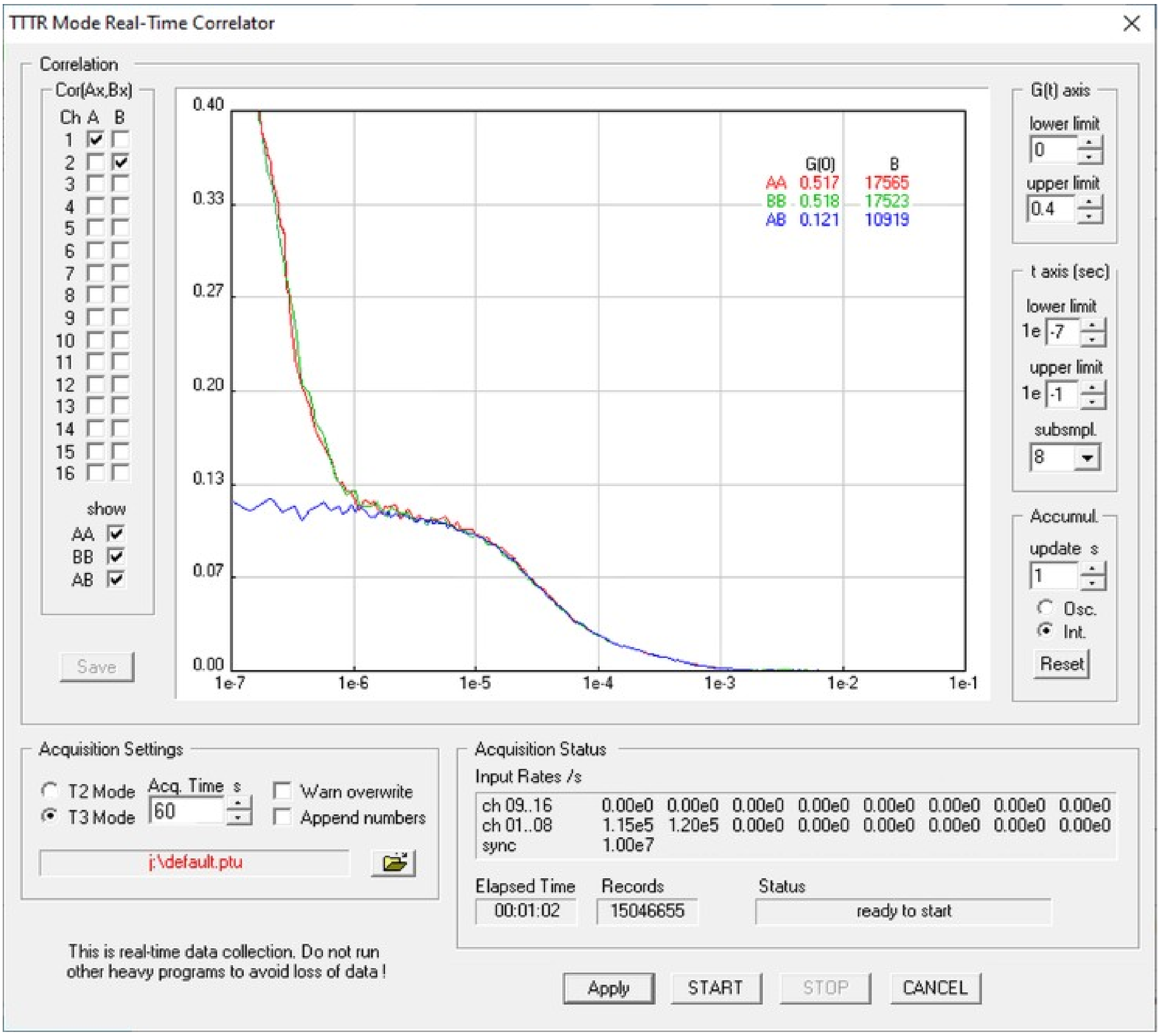
The lower section of the dialog provides the same file handling and progress indicator elements as in regular TTTR mode. Please refer to section 6.3 for details. The upper section of the dialog contains the correlator display and various control elements. The dialog section *Cor(Ax,Bx)* allows to select individual input channels for the correlation. Note that this is limited to 16 channels, even when the hardware has more input channels. A and B are the virtual input channels to be correlated. All physical input channels that are ticked in columns A and B, respectively, will be aggregated and used in the corresponding correlation channel. If your selection is empty, the heading of the section *Cor(Ax,Bx)* will turn red to indicate the problem. If you proceed anyway you will get error messages upon start.

The correlator always calculates the autocorrelations of the two virtual channels (AA and BB) as well as the cross-correlation (AB). There are three tick boxes that select which of these are shown.

The dialog section *Accumul.* allows selecting an update time for the correlation display. Furthermore it allows to select between repetitive updates (*Osc.*) and cumulative collection (*Int.*). If the latter is chosen one can manually reset the correlator by clicking the *Reset* button. Note that all these settings affect only the real-time correlator. The raw TTTR mode data will be collected continuously and completely independent from the correlator settings.

The *Save* button under the *Accumul.* section allows to save the correlator results as they were last shown. The saved result is an ASCII file with some header information. The format is self-explanatory.

Note that the real-time correlator is primarily a tool for preview during measurement setup and optimization. It does not allow to correlate data from TTTR files. If you need the correlation over the entire stream of collected data you need to make sure that the measurement runs in Integrating mode and that you do not press reset during the acquisition. If you need the correlation results you should save them here. If you need to perform further or more thorough FCS analysis on the collected data you can use the [SymPhoTime 64 software] from PicoQuant.



On the right hand side there are controls for the axis ranges of the correlator display. Note that they affect only the display. Collected data is always complete, independent from the axis settings. In contrast, the selection box *subsmpl.* has an effect on the correlator results. It determines how many tau points are calculated. The correlator works in a logarithmic multiple tau scheme and *subsmpl.* specifies the number of linear subsamples in each log stage. A higher number of subsamples increases the resolution of the correlation curve. Calculating more points is more time consuming and therefore may lead to lower count rate limits that can be handled. In this case you may get FIFO overruns. The default of subsmpl.=8 is a reasonable trade–off between speed and resolution. On faster computers the setting of subsmpl.=16 is a good choice.

Note that the starting point and spacing of the tau sampling points also depends on the time tag resolution. In T3 mode this corresponds to the sync period. In T2 mode the native resolution of the board is binned down to a time tag resolution of 25 ns in order to make the data manageable for the real-time correlation algorithm.

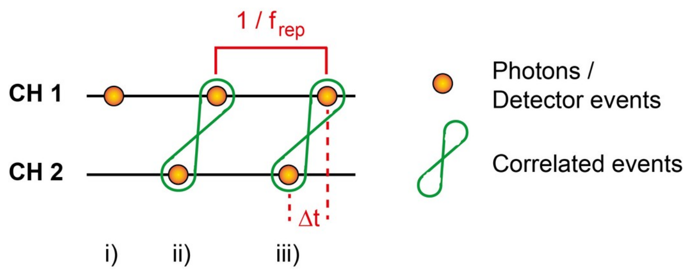
The correlation curve display can be shown with a grid (see general settings dialog, accessible from the toolbar of the main window). It also shows two useful figures obtained from the collected data (top right of curve window). The first figure is an approximation of G(0). In classic FCS experiments this corresponds to the inverse of the number of particles in the focal volume. It is continuously updated together with the curve display, which can be useful for system adjustments, notably when using the repetitive accumulation mode (Osc.). Note that the approximation is a simple averaging over the first ten tau points. The figure B is an indicator for molecular brightness. It is also updated continuously. Note that it is also only an approximation obtained by multiplying the G(0) approximation with the average count rate on both virtual correlation channels A and B. Depending on the chosen channels in A and B this may lead to figures that do not truly reflect molecular brightnesses. However, they should be useful as an adjustment aid in any case.

All other aspects of TTTR data collection with correlator preview are the same as in plain TTTR mode as described in the previous subsections. Topics such as using external markers or how to use the data files should be looked up there.

6.3.8. TTTR Mode Measurements with Event Filtering

Introduction and Application Context

In many quantum optical application scenarios, information is encoded in the correlation of two or more photons. Traditionally, coincidence measurements are used to evaluate these correlations experimentally. These range from relatively simple experiments like bipartite entanglement measurements to more complex setups as used in sampling or universal quantum computation experiments. The following figure sketches a stream of correlated photon pairs across two detection channels, as for example generated by spontaneous parametric down-conversion (SPDC) pumped with a pulsed laser of repetition rate f_{rep}.



Due to optical and electrical delays the correlated photons may arrive at the detectors at different times. This creates a time offset Δt between the event in detection channel CH1 and the event in detection channel CH2 that belong to one correlated photon pair (green loops at time positions ii and iii).

Historically, early detection hardware like coincidence logic analyzers, required that all signals needed to be temporally well synchronized. This corresponds to negligible Δt in the sketch above and the correlated events in CH1 and CH2 happen simultaneously – they coincide temporally. A coincidence count is then just the logical combination of these two detection events. In coincidence logic analyzers this combination is carried out in hardware. In order to allow for small tolerances it is good practice to allow for a range of temporal offsets Δt that a coincidence analyzer still counts as coincidences – this range is often called coincidence time window. Using modern time tagging TCSPC systems, there is no need for an a priori synchronisation of the detector channels. In such systems, the time offset between the channels can be arbitrarily shifted in hardware and/or postprocessing. Consider a time offset Δt even larger than the one shown in the sketch above. Here, a photon from an earlier pair in CH1 (i – no green loop) would falsely be aligned with a photon from a later pair in CH2 (ii). Counting this event as a coincidence would not allow to reveal the true correlations. This examples illustrates the advantage of modern TCSPC systems: when using a hardware coincidence logic analyzer the whole dataset of the measurement would be lost in the case of a wrong delay alignment. With modern time tagging TCSPC systems the correct correlations can be easily recovered by adjusting Δt during data analysis.

However, coincidence logic analyzers also have their unique advantage. In the sketched quantum optics application scenario, the correlations of interest only exist between pairs or higher numbers of photons. Often lower-order events, e.g., single events in the case of two-fold coincidences contain little information for these experiments. Coincidence logic analyzers already discard these lower-order events and hence reduce the recorded data size, often drastically. This does not only make data analysis less cumbersome and file size smaller, but also allows for higher data rates for the signals of interest, as the bus interface to the host PC is not burdened with unnecessary data. Hence, data can be processed and displayed on a host PC in real time.

The event filter combines the advantages of TCPSC devices with the advantages of traditional coincidence logic analyzers. In order to make the filter beneficial for a wide range of quantum optical application scenarios it is designed to operate on all input channels of the HydraHarp 500 in use and hence the full set of their possible combinations. Traditional coincidence logic analyzers often predefine logical combinations between channels (coincidence patterns) to be recorded, while all other events are discarded. This may limit applications as the set of predefined patterns is usually hardware limited. The HydraHarp 500 's event filter strives to resolve this by defining discardable orders of events (e.g. single events) and keeping what may be of interest. This reduces the data load sufficiently for lossless bus transfer and reasonable file size. Computing logical combinations between channels in the remaining data can then be easily done on the host PC at virtually unlimited complexity.

In the simplest setting, the filter discards all events that do not have partners (in any channel) within the coincidence window of programmable width. This already permits a significant data reduction without loss of generality for any subsequent software analysis. For further reduction the order of filtering (the number of partners) can be set to higher number by way of a parameter “match count”, so that only 2-fold, 3-fold,…, up to 6-fold coincidences are recorded.

In contrast to many traditional coincidence logic analyzers and most of the current time tagging systems, all raw events of the contributing channels passing the filter are fully recorded with their time tag. Their complete information content is therefore available for refined analysis in subsequent software processing. Setting the time window (set by the filter parameter ‘range’) can therefore not only act as an immediate coincidence time window, but initially can also be set broader for a first mild data reduction. As all events are precisely time tagged, a narrower coincidence time window (or multiple such windows) can always be applied in post-processing. The “range” setting up to a certain degree tunes how close the system will behave to a plain TCSPC or time tagger without event filter and, in the opposite, a generalized coincidence logic analyzer. Note, that for the hardware event filter to work properly, the channels need to be temporally aligned. This is easily achieved by means of the programmable offset in each of the HydraHarp 500 ‘s input channels.

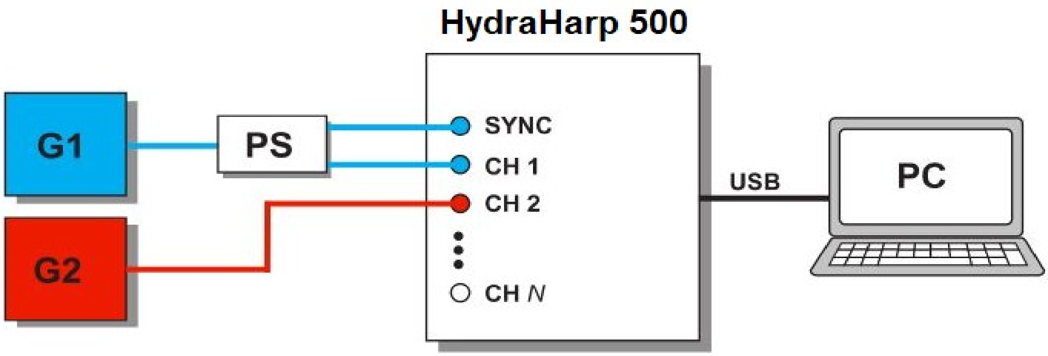
In addition to the filter functionality outlined so far, all event timing channels can optionally be marked as ‘pass channels’ that will be passed through the filter unconditionally. This can be very useful to provide reference channels, e.g., if the channel counts contain information to track drift of the setup in the same measurement.

As a further means of flexibility and for test purposes the filter provides an optional ‘inverted’ setting. In that case the filter discards all events that do have partners according to “match count”, e.g., events that have 2, 3, .., 6 partners are then discarded.

To summarize: In many quantum optical experiments it is typically of little interest to collect photon events that occur without one or more photons in close temporal context. Recording such superfluous events does no harm to the experimental objectives as they can be filtered out in software post-processing. However, it unnecessarily consumes bus bandwidth and bloats the recorded data files. The HydraHarp 500 in particular has an extremely high input bandwidth: each channel can in principle capture events at up to 1.25 GHz. Despite a sophisticated FIFO buffering scheme (see section 6.3.2) combined over the instrument’s many channels this high rate cannot be sustained for long in internal processing and USB transfer. It is therefore of interest to reduce the data rate at an early stage directly in the hardware. This objective is achieved by the HydraHarp 500 ‘s event filtering mechanism.

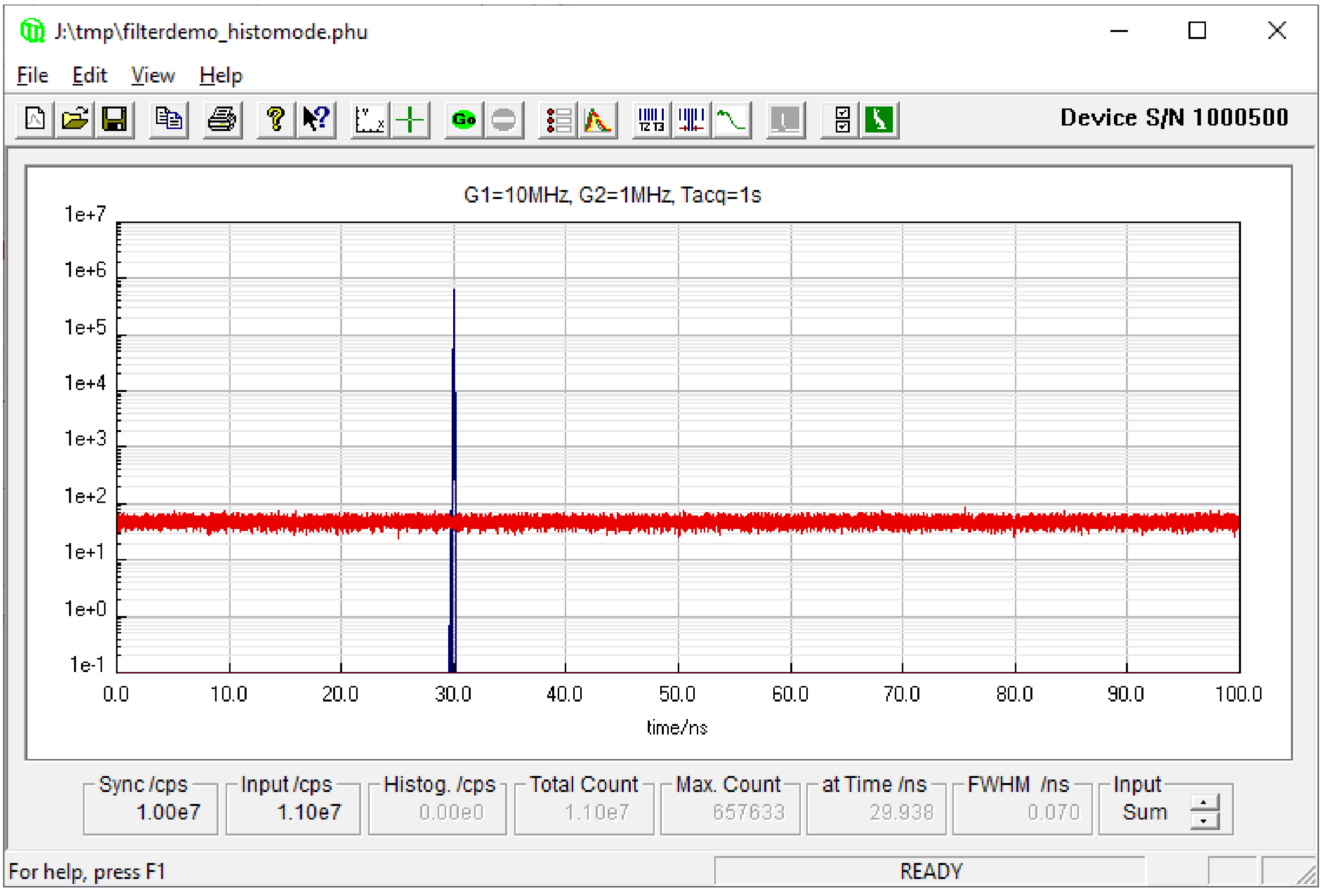
Event Filter Usage Walk-Through

In the following we discuss a simple experiment showing how the filter works and how it can be programmed. The experimental setup we will be using throughout this section is shown in the figure below. For the sake of clarity and repeatability we use signal generators of fixed frequency rather than the mostly non-deterministic signals found in quantum optics.

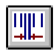


Generator G1 provides pulses at a frequency of 10 MHz. By means of a 50 Ohms power splitter (aka reflection free T-pad) identical pulse trains are fed to the HydraHarp 500 ’s sync input and photon timing input channel 1. Generator G2 provides an independent periodic pulse train of 1 MHz that is fed to the HydraHarp 500 ’s photon timing input channel 2. The generator frequencies and the choice of input channels are somewhat arbitrary and we only pin them down here for consistency with instrument settings and experimental results shown further on.

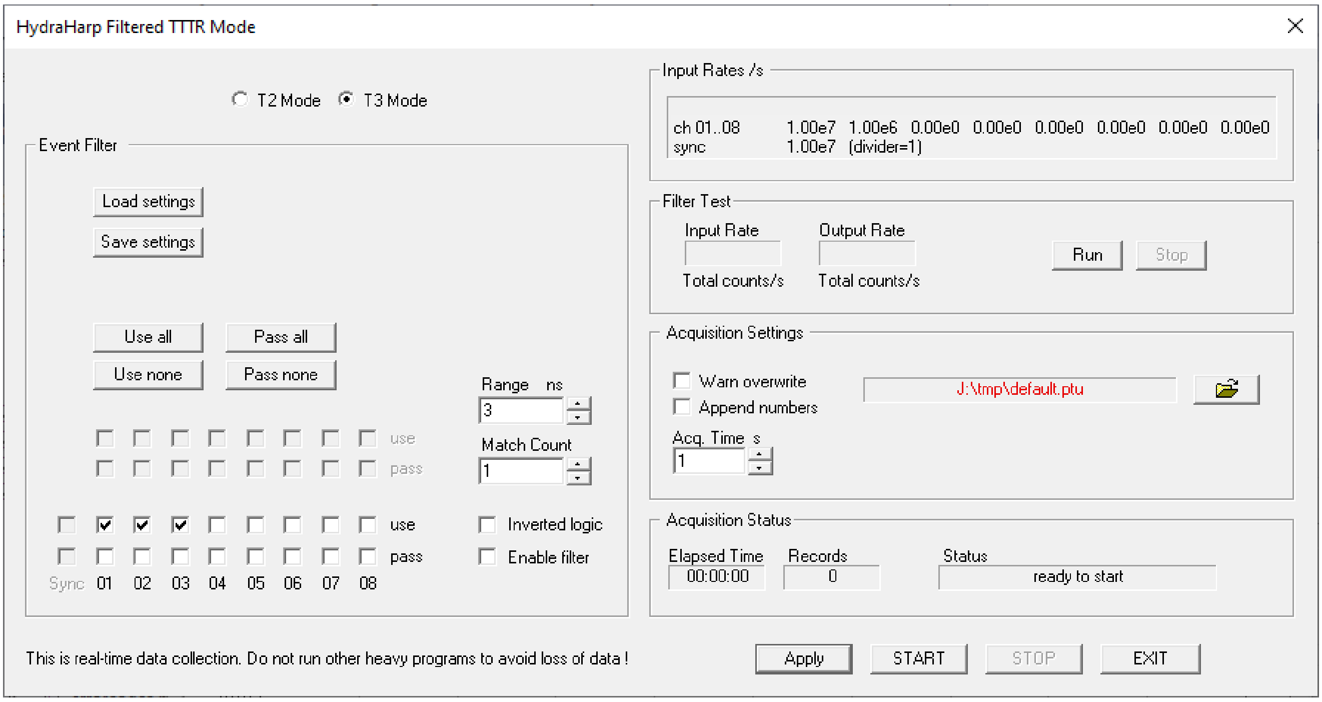
While event filtering is a feature available solely in TTTR mode, it is helpful to first consider what kind of results they would generate in histogramming mode. Indeed the experimental setup was designed to permit this and for didactic purposes you are encouraged to try it out. Given that the signals from G1 are identical on sync and input channel 1, the time difference between them is constant, except for a small stochastic measurement error of a few ps. Therefore, one would expect a histogram looking like a sharp spike. With identical cable lengths after the splitter we should expect the peak to sit at time zero of the histogram axis. Since the histogram results from positive timing differences between input channel and sync, we would only be able to see the positive half of the peak. In order to see the full peak it is useful to introduce a delay in the input channel. This can be done physically by means of cable, but one great feature of PicoQuant’s TCSPC systems is that the same can be achieved by means of a programmable offset, so we use the latter and set it to 30 ns. The signal from G2 is temporally independent from G1 so that the corresponding time difference is taking all possible values within the sync period, i.e. the 100 ns period of G1. Over some sufficient measurement time we would therefore obtain a histogram that is evenly filled, with only some statistical fluctuations according to counting statistics. The figure following below shows the actual measurement results.



Now we take one step further and perform the same experiment in TTTR mode, first without filtering and then with filtering. We use T3 mode because it is functionally related to histogramming mode, in that it also assumes a typically periodic sync signal and, more important here, the HydraHarp 500 software permits loading T3 mode files directly into the histogram display. We could do the experiment by way of a regular TTTR mode measurement (See section 6.3.4) but for didactic purposes we shall do it through the dialog for Filtered TTTR mode.



The Filtered TTTR Mode dialog can be reached through the corresponding toolbar button shown here on the right. The dialog is conceptually similar to that of regular TTTR mode. Upon clicking on the button you will notice a moment of delay where the hardware is re-configured for the new measurement mode. When this is completed the dialog will appear similar to the figure following below.



Most prominent on the top left there is a pair of radio buttons for T2 versus T3 mode. It is important to make this selection first, as it will determine the permitted filter settings for the sync channel.

Below on the bottom left in the group box ‘Event Filter’ there is a matrix of tick boxes for the selection of whether and how individual channels are going to be used in the filter. Note that this matrix caters for the largest possible HydraHarp 500 device with currently up to 16 input channels. If your particular device has fewer channels, the extraneous elements will be disabled and grayed out. For each channel there is one tick box ‘use’ to indicate if the channel is going to participate in the data collection and in the filtering scheme. In addition, for each channel there is another tick box ‘pass’ to indicate if the channel is to be passed through the filter unconditionally, whether it is marked as ‘use’ or not. The events on a channel that is marked neither as ‘use’ nor as ‘pass’ will not pass the filter, provided the filter is actually enabled, as marked by the tick box ‘Enable filter’ further right on the bottom.

Also within the group box ‘Event Filter’ there is a set of buttons for loading and saving the filter settings. This is a convenience feature for quick retrieval of standard settings you may wish to keep. The filter settings are also memorized in the Windows registry as well as in each data file. Yet another convenience feature are the buttons ‘Use all’, ‘Use none’,‘Pass all’, and ‘Pass none’. The make it easier to fill entire rows with the desired setting.

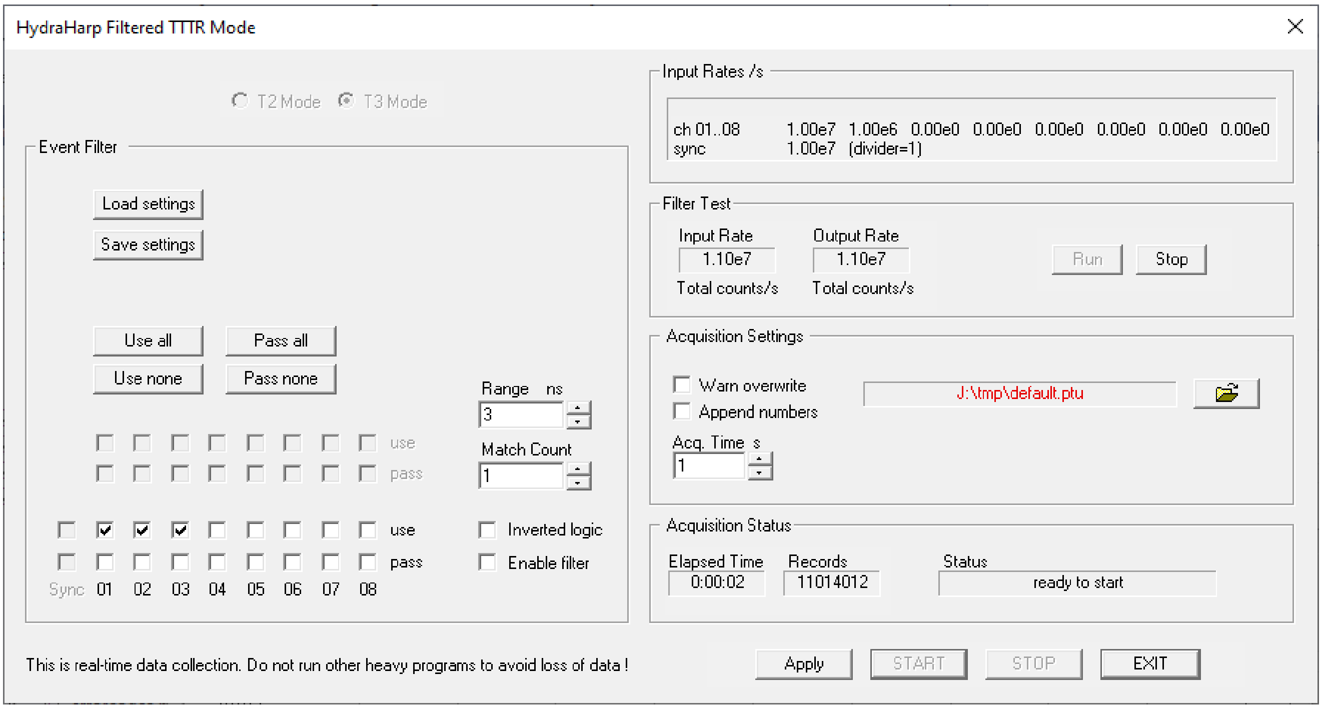
The other controls ‘Range’, ‘Match Count’, and ‘Inverted logic’ determine how the filter will act. ‘Range’ determines the time window the filter is acting on. It can be entered in whole nanoseconds or as floating point numbers. If you enter fractions smaller than the device’s resolution the effective range will be rounded down. Note that Range acts both ways in time so that the window width is actually 2xRange. The parameter ‘Match Count’ specifies how many other events must fall into the chosen time window for the filter condition to act on the event at hand. The tick box ‘Inverted logic’ inverts the filter action, as you will see more clearly further in this section. For now, let it be not inverted. Then, for instance, if ‘Match count’ is 1 we will obtain a simple ‘singles filter’. This is the most straight forward and most useful filter in typical quantum optics experiments. It will suppress all events that do not have at least one coincident event within the chosen time range, be this in the same or any other channel marked as ‘use’.

Before showing the actual filter operation, let us first complete the introduction of the remaining dialog elements. The group box ‘Input rates’ at the top right is self explanatory and identical to what you find in the regular TTTR mode dialog. It gets updated every 500 ms.

Further down on the right you find a group box ‘Filter Test’. This provides a useful feature for testing a filter configuration, and in particular checking as to whether the resulting data rate may or may not be sustainable over USB. Recall that the HydraHarp 500 has an extremely high input bandwidth: each channel can in principle capture events at up to 1.25 GHz. Despite a sophisticated FIFO buffering scheme (see section 6.3.2) combined over the instrument’s many channels this high rate cannot be sustained for long in internal processing and USB transfer. The filter test will permit to try this out without doing a real measurement and running into FIFO overruns. Because the test performs a quasi-real measurement, only without forwarding data from the filter to the

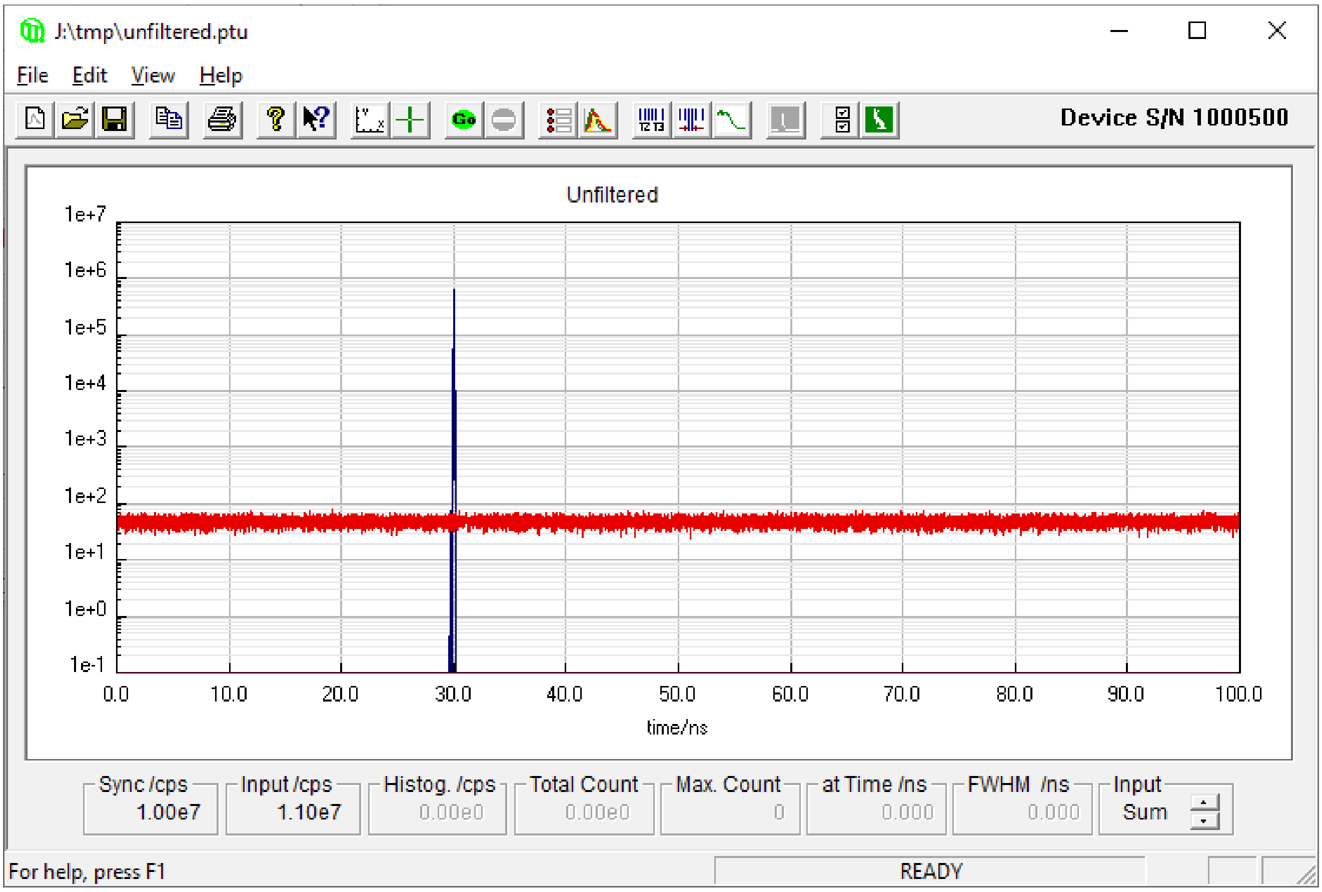
FIFO, it cannot be performed when a real measurement is running and vice-versa. The corresponding buttons will be mutually disabled and greyed out to prevent this.

Using the setup with the two generators as sketched above we can now perform a filter test by clicking the Run button and observe the Input and Output rates:



As expected with the filter still being disabled, the Output Rate matches the Input Rate.

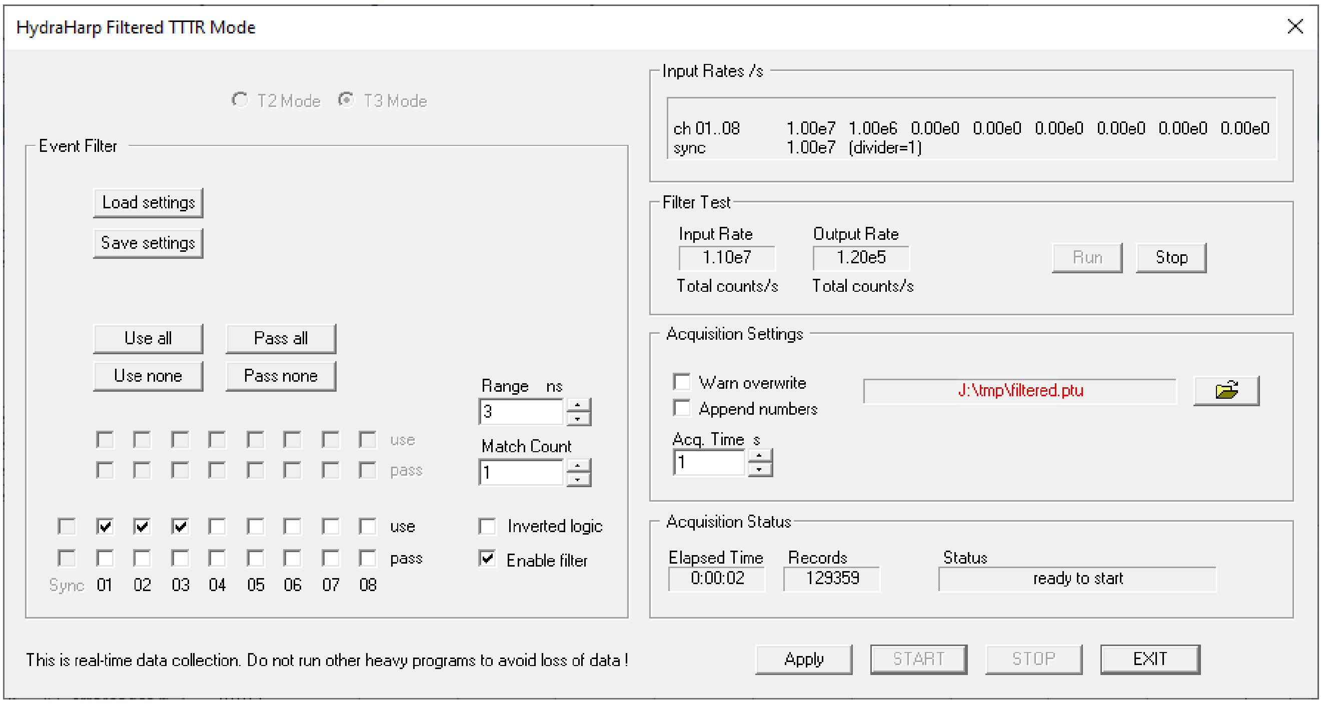
After stopping the filter test we can then proceed to a real measurement. This involves the dialog elements for 'Acquisition Settings' and Status as well as the buttons START and STOP. The procedure is the same as in plain TTTR mode described in section 6.3.4. We use an acquisition time of 1 s and the filename 'unfiltered.ptu'. After completion of the measurement we leave the filtered TTTR mode dialog by clicking EXIT. When the software is back in histogramming mode we load the file we have just collected into the regular histogramming mode display. Note that this requires selecting the file type *.ptu in the file opening dialog. As the collected data is from T3 mode, the software will prompt for which range of the TTTR records it should perform the histogramming. We accept the default setting of using the entire range. The result will be a set of histograms as shown in the next figure:



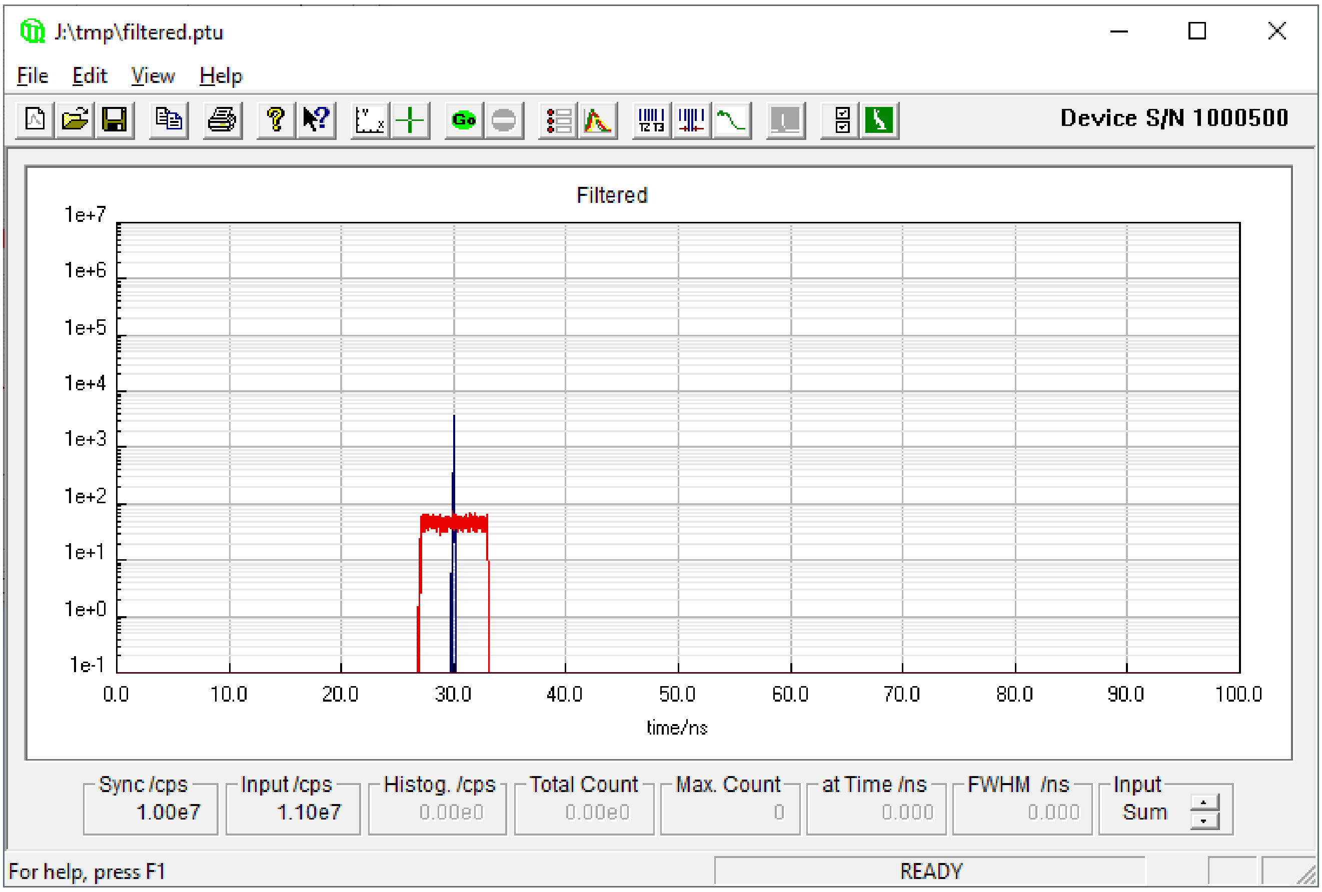
As expected, because the filter was disabled, we obtain histograms exactly as in our initial test in histogramming mode.

Having completed this basic test we can now return to the filtered TTTR mode dialog and proceed to a filtered measurement. The filter settings are the same as previously, except that the filter is now enabled.

Like before, we perform a quick filter test. As we see in the next figure, the output rate has now dropped to 1.20e5.



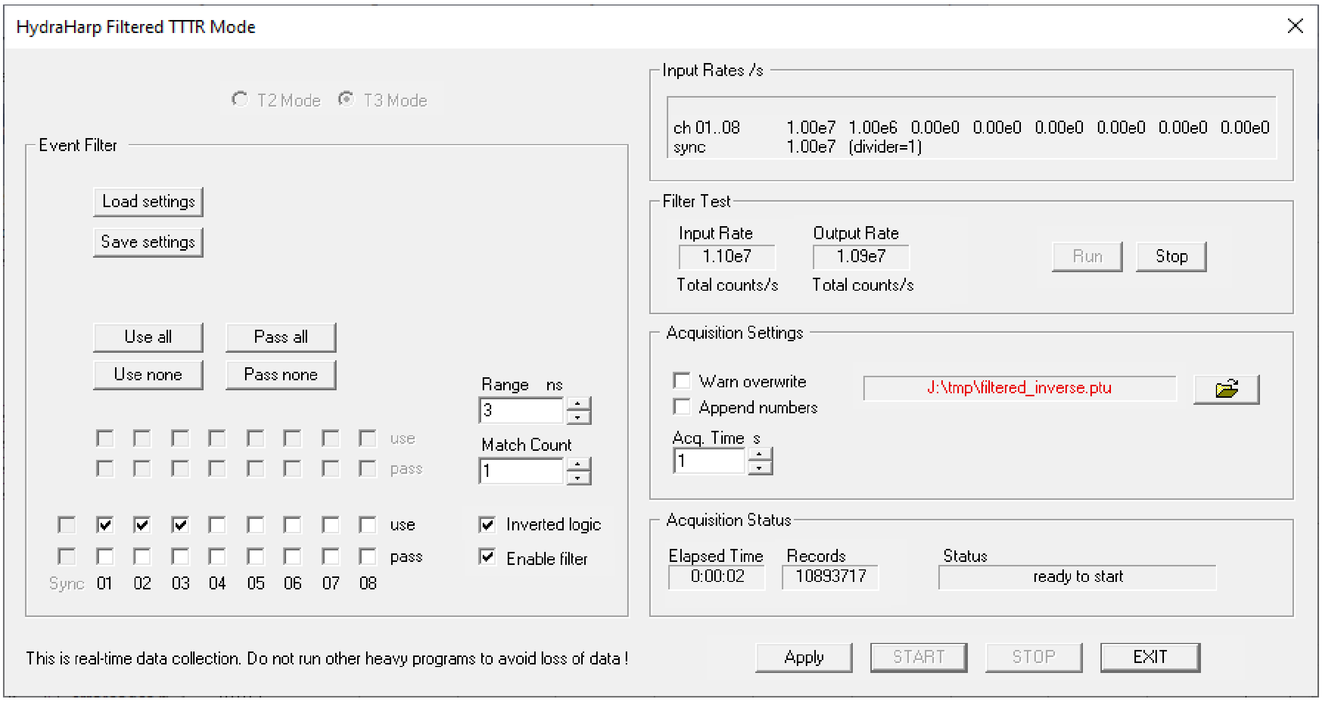
After stopping the filter test we can again proceed to a real measurement. This time we name the file 'filtered.ptu'. After completion of the measurement we can again leave the filtered TTTR mode dialog and then load the new file into the regular histogramming mode display. As this switching between TTTR and histogramming mode always takes time for re-configuration you may want to use a little trick: If you open a second instance of the HydraHarp 500 software you can leave the first instance in TTTR mode and use the second instance as a file viewer. Whichever way you load the file, the result will be a set of histograms as shown in the next figure:



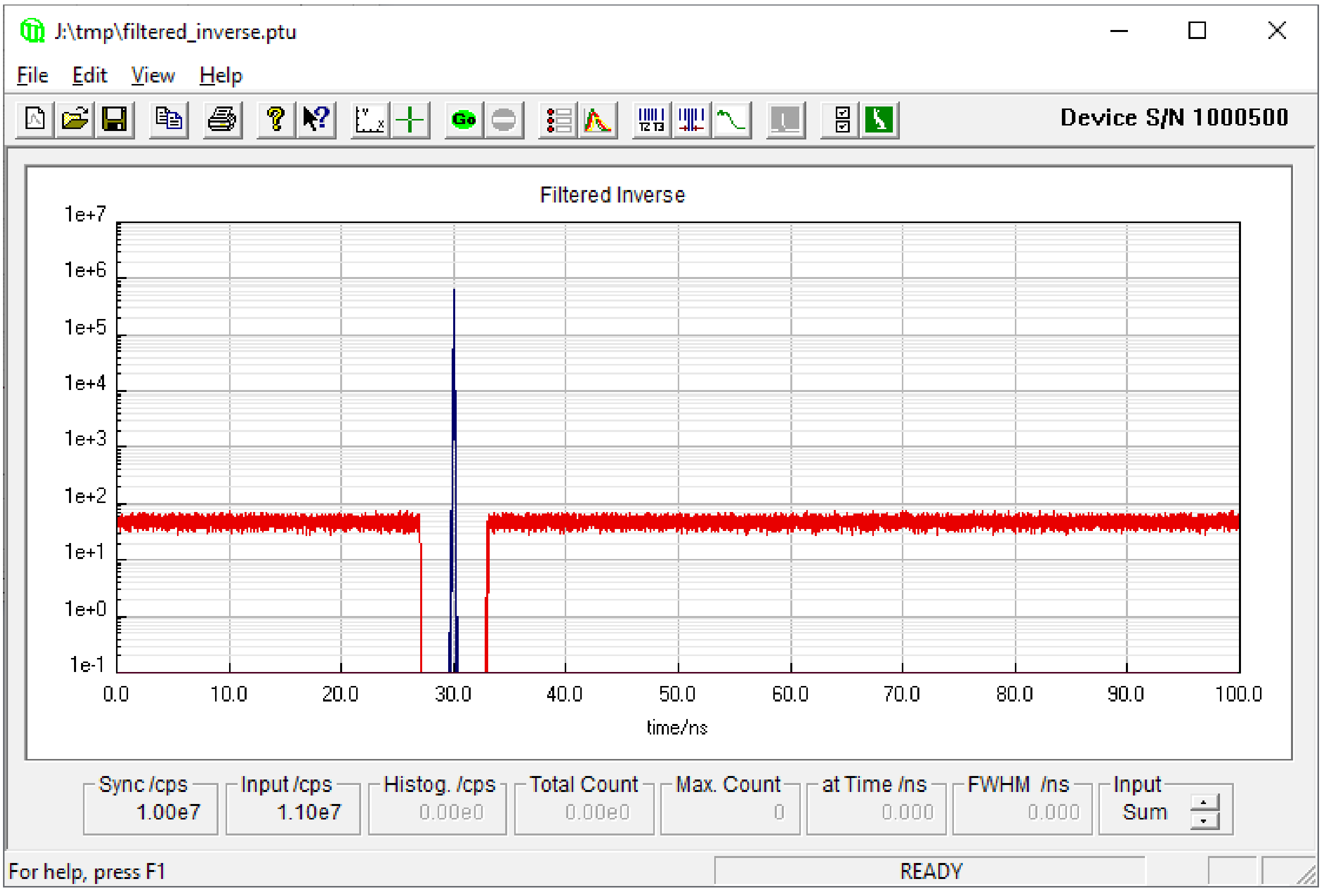
The effect of the filter can now nicely be seen. First observe the red trace (Generator G2 on Channel 2). The histogram is now empty everywhere, except in the vicinity of the blue peak. The remaining slice of the red trace is exactly the chosen filter range of 3 ns on either side of the blue peak from G1. Next, observe the blue peak. Compared to the unfiltered data collection it is now substantially smaller. This also is the effect of the filter, as all blue events that had no red companion in the 3 ns range were suppressed. This now gives a nice visual explanation for the significantly lower output rate of the filter we observed before.

Finally, we can perform the same experiment with the filter switched to 'Inverted logic', leaving all other settings as before. What this means is that the filter will now suppress events that previously would NOT have been suppressed and vice versa. This provides a nice way of filter verification and demonstration.

The following figure shows the corresponding state of the filter dialog with filter test running. The output rate is now significantly closer to the input rate but from the fact that it is lower than the input rate we see that some events are still being filtered out. Indeed you may check that the filter output rates of inverted versus uninverted logic sum up to give the input rate.



Performing a real measurement into the file 'filtered_inverse.ptu' and loading the file into the histogram display we can nicely visualize the inverse filter operation:



The part of the red histogram that was previously kept in is now punched out and the blue peak is diminished much less, according to the ratio of removed versus unremoved events in the red histogram.

This concludes our demonstration of the filter operation but some more practical hints shall be given here:

It is possible and quite useful to keep the filter test running while playing with the individual filter settings and observing the output rate. However, any change of filter settings must always be confirmed with Apply in order to take effect.

In some application scenarios it may be worth considering another type of filter. In fact at hardware level there are two types of filters: The Group Filters and the Main Filter. The graphical HydraHarp 500 software described here implements only the latter. This is mainly for simplicity, because only the Main Filter can act on all channels as well as the sync channel simultaneously. In order to achieve this, it needed to be implemented in the main FPGA that collects and merges all the data from the individual measurement modules before transmitting it over USB. The drawback of this is, that an overload of data rate within the individual measurement modules cannot be prevented this way. Although it is unlikely that many applications will run into this limit we also implemented the Group Filters, each acting on a group of 4 channels (HydraHarp 500 S) or 8 channels (HydraHarp 500 M). These can also be useful if you require more complicated filter conditions, e.g., with multiple different time ranges. In that case the two levels of filters can be daisy-chained. However, in order to do this you will have to use either the new UniHarp software, or the high-level API package “snAPI”, or the HydraHarp 500 programming library and your own custom software. If the problem is not one of group-wise data rate overload it may be easier and more flexible to use only the main filter and to perform the more complex filtering in software.

All other aspects of TTTR data collection with event filtering are the same as in plain TTTR mode as described in sections 6.3.1 through 6.3.6. Topics such as using external markers or how to use the data files should be looked up there.

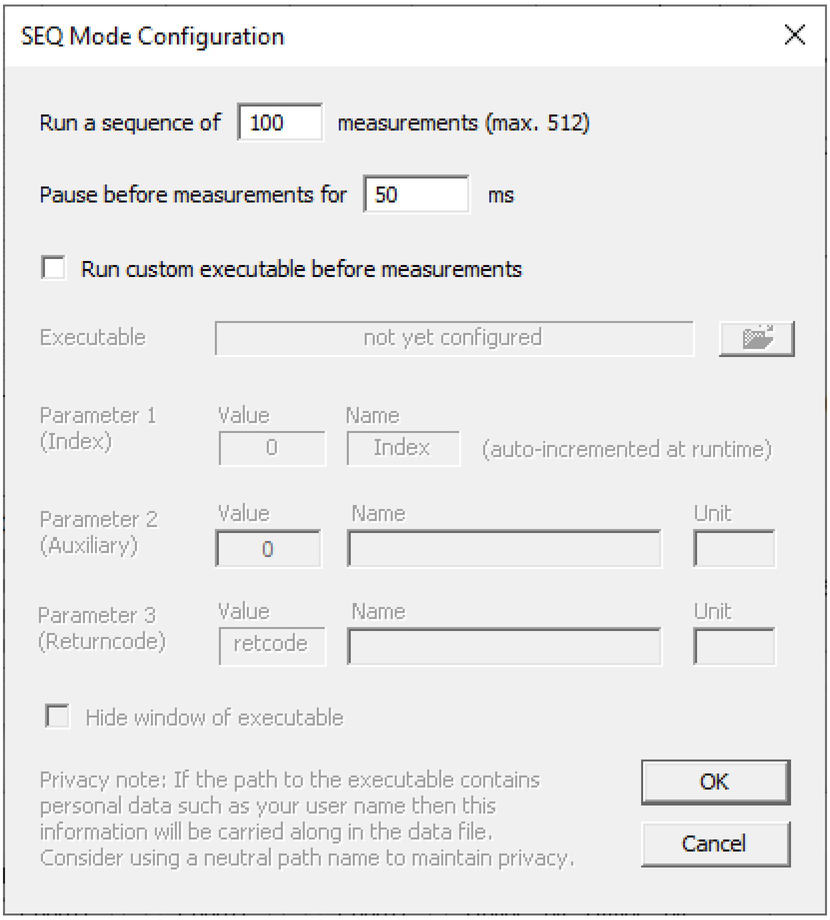
6.4. Sequence Mode

In time-resolved fluorescence research it is often of interest to observe time-dependent phenomena. In the simplest case one would then collect a sequence of decay histograms. A similar requirement arises for recording Time-Resolved Excitation/Emission Spectra (TRES), ideally with automated wavelength scan and TCSPC data collection under full software control.

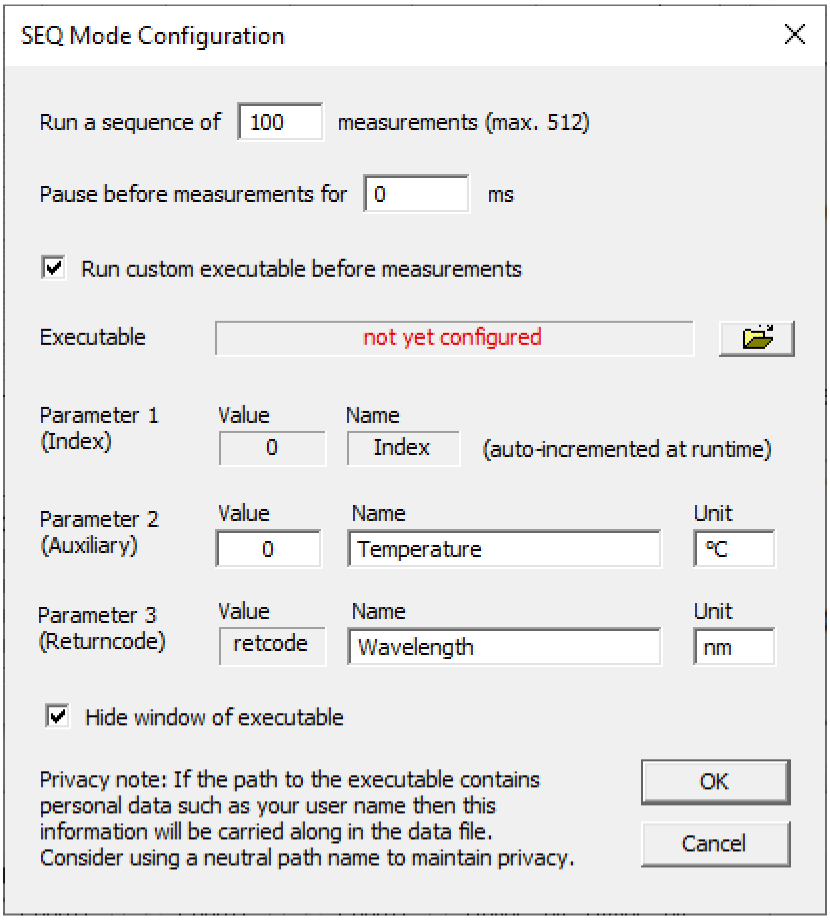
In order to facilitate a generic range of such applications with great flexibilty, the HydraHarp 500 software provides an automated sequence (SEQ) measurement mode. This mode first of all allows software controlled repetition of a histogram mode measurement. For more complex automated experiments it also allows calling a custom piece of software before each measurement. This can be a Windows executable, a batch file or even a Python script. By way of this custom code the users can control their own external hardware, for instance a monochromator. This enables automated collection of spectrally resolved lifetime histograms. Similarly some other parameter such as temperature could be stepped through. SEQ mode data is collected as in standard “Integration Mode” and saved in different blocks of memory for each cycle. Note, however, that SEQ mode data is always collected through input channel 1 only.

SEQ mode can be configured by clicking the corresponding button on the toolbar: 
This will launch the dialog for SEQ mode setup.

There you can first of all set up the length of the sequence (the number of measurements) and optionally also the duration of a pause between the measurements.



If then below you select (tick) the tickbox “Run custom executable before measurements” you will see further dialog elements getting enabled. The topmost row in this category holds the path name of the executable. It will appear in red if it is not yet configured or invalid. In order to select an executable just click the “file open” button on the right and browse to a suitable file.



The executable, batch file or Python script must be written for the specific purpose and it must follow some rules. The fundamental idea can be illustrated looking at the next three rows of configuration elements called Parameter 1 … Parameter 3, some of which can be edited. Parameters 1 and 2 are going to be passed to the custom executable. Parameter 1 is simply the zero based index of the measurement. It will automatically increment and cannot be edited. Parameter 2 is an auxilliary value that can be edited and will also be passed to the executable. Its value will remain fixed throughout the sequence. This can be used to parametrize the behavior of the executable without having to re-compile it or editing the script. Parameter 3 is the return code from the executable. All three parameters will be written to the header of the corresponding data curve in the SEQ mode file. If, for instance, the executable was built to control a monochromator, it could be designed to return the wavelength of this measurement. By definition a negative return code will indicate an error and the run of the sequence will be aborted. Because of constraints in Windows the return code can only be a 32-bit integer. Demo code fo an executable in C, a batch file and a Python script can be found in the subfolder SEQdemo under the path where you installed the HydraHarp 500 software.

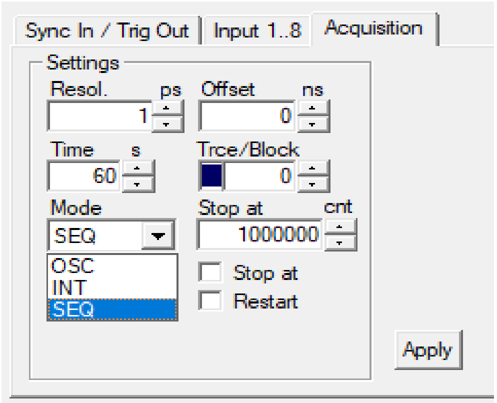
The fields *Name* and *Unit* can be populated with custom strings in order to embed some documentation of the measurements in the file. For similar reasons the path of the executable will be embedded in the file.

Further below in the SEQ configuration dialog you will find another tick box “Hide window of executable”. When it is not ticked, the executable’s window will pop up each time it is called, i.e. for each step in the sequence. This can be useful for debug purposes but it may also become annyoing. In that case the box for hiding it should be ticked. The executable’s window will then not pop up.

Running SEQ Measurements

Note that starting a SEQ measurement will clear and subsequently overwrite all curves in memory that may have been recorded previously. Note also that a SEQ file will remain a SEQ file for its lifetime. This means that after creating or re-loading such a file you can do only two things: perform a new SEQ measurement (effectively overwriting the old file contents) or measure an IRF into curve 0 (overwriting only curve 0). If you want to perform any other measurement you will need to create a new file or load a non-SEQ file.

The SEQ measurement is selected in the HydraHarp 500 control panel. You need to use the drop-down box for the measurement mode to select SEQ mode (instead of OSC or INT).

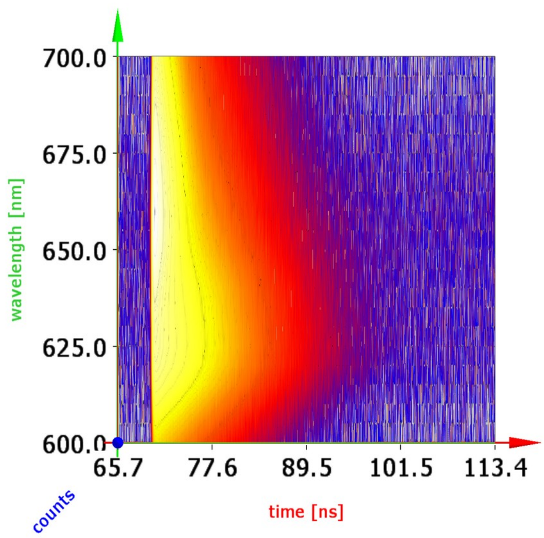


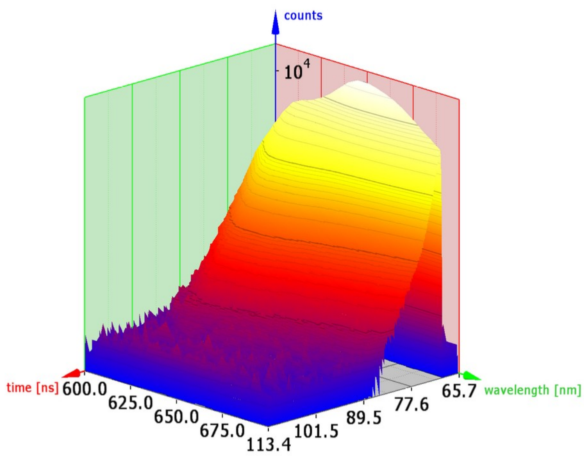
A SEQ measurement can then be started or canceled as usual via the toolbar buttons for start and stop. The data acquisition will be performed like a standard “Integration Mode” measurement, while the block number is incremented for each new cycle.

At the beginning of each SEQ run, the active curve (block) will be set to 0. All curves that were previously collected and not saved to a file will be overwritten with new data. During a SEQ run, the currently collected data will always be shown as Trace 0 (dark blue), even though the block (Trace) number is actually incremented at each step. This is to overcome the limitation of only 16 display curves being available. You can check the block number in the control panel to see the curve currently being collected. Manual entry will be disabled during the run. Also, you can watch the status bar to see the current activity. Note that for a SEQ measurement the control panel setting 'Stop at' applies as in Integration mode, while 'Restart' is meaningless in SEQ mode.

SEQ Data Analysis and Visualization

Having collected a complete SEQ data set you can save it to a regular `*.phu` file. You can also inspect individual curves via the Trace Mapping dialog. You can furthermore use the software tool FluoPlot from PicoQuant to visualize and analyze the data in various 2D and 3D representations with a multitude of options for coloring, scaling, and changing view aspects in 3D. The figures below show FluoPlot visualizations from a time resolved emission spectra measurement of mixed oxazine dyes in ethanol.





FluoPlot is provided as part of your HydraHarp 500 software download. Just run FluoPlotSetup.exe from the unpacked zip file. Note that the program requires a graphics processing unit with support for OpenGL version 1.5 or higher. Speedy handling of large data files furthermore requires sufficient system memory. Further data analysis may need to be performed by dedicated software, either custom programs or specialized solutions available from PicoQuant.

6.5. Multi-Channel Scaling

In contrast to classic TCSPC systems (based on TACs), the sync and the input channels of the HydraHarp 500 are completely independent. This makes it possible to allow multi-stop measurements, i.e. the detection of multiple photons between two subsequent start (sync) signals. The shorter the dead times of the detectors and the timing electronics are, the less these multi-stop measurements are prone to dead-time-induced photon losses. If the histogram bin width is chosen wider than the dead time, the pile-up artifacts due to dead time are entirely eliminated and it is possible to measure at photon rates much higher than the classic pile-up limit of TCSPC. The HydraHarp 500 can then be operated like a Multi-Channel Scaler (MCS).

The easiest way to perform multi-channel scaling with the HydraHarp 500 is histogramming mode. The detector is connected to one of the input channels. In addition, a reference signal at the sync input marking the start of the MCS measurement is needed (see chapter 9.3.1 for the signal specifications).

The integration time should then be set to a value significantly larger than the time bin width multiplied by the number of time bins (65536). Then start a measurement in in integration (INT) mode. The resulting histogram will display the MCS curve. Please note that the time range that can be obtained with this method is limited to the time bin width multiplied by 65536. MCS measurements on longer time ranges are possible via HydraHarp 500 programming library (131072 time bins in histogramming mode) or virtually unlimited via T2 mode. However, this will require additional software for data analysis (e.g., SymPhoTime 64 or snAPI).

7. Controls and Commands Reference

7.1. Main Window

The Title Bar

The title bar is located along the top of a window. To move the window, drag the title bar. Note: You can also move dialog boxes by dragging their title bars. The title bar may contain the following elements:

System Menu	
HydraHarp 500 – [Name of the file or dialog]	
Minimize Button	[–]
Maximize or *Restore* Button, resp.	[□] / [⧉]
Close Button	[X]

Scroll bars

Displayed at the right and bottom edges of the window if a certain minimum window size is reached. The scroll boxes inside the scroll bars indicate your vertical and horizontal location in the display area. You can use the mouse to scroll to other parts of the window.

Size command

Use this command to resize the active window.

Note: The command is unavailable for already maximized or minimized windows.

Shortcut

Mouse: Drag the corners or edges of the window.
Keys: <Alt>+<Space> S

Minimize command

Use this command to reduce the HydraHarp 500 window to an icon. Running measurements will continue.

Note: The command is unavailable for already minimized windows.

Shortcut

Mouse: Click the minimize button [–] on the title bar.
Keys: <Alt>+<Space> N

Maximize command

Use this command to enlarge the active window to fill the available space.

Note: The command is unavailable for already maximized windows.

Shortcut

Mouse: Click the maximize button [□] on the title bar;
or double–click the title bar of a non-maximized window.
Keys: <Alt>+<Space> X

Restore command

Use this command to return from a maximized or minimized window to the previous size.

Note: The command is only available for already maximized or minimized windows.

Shortcut

Mouse: Click the maximize button on the title bar;
or double–click the title bar of the enlarged window.

Keys: `<Alt>+<Space> R`

Close command

Use this command to close the active window or dialog box.

Double–clicking a system menu box is the same as choosing the Close command.

Shortcuts

Mouse: Click the close button on the title bar.

Keys: `<Alt>+<F4>`
`<Alt>+<Space> C`

Panel Meters

At the bottom of the main window there is a set of panel meters. The meters showing units of cps (counts per second) in their title are rate meters. The leftmost rate meter shows the sync input rate. The next meter shows the channel input rate. Note the selector on the right of the panel meters. This selects the channel the rate meters (except Sync) are referring to. The other meters show the histogramming rate, the total count in the histogram, the maximum (peak) count, and the position of the maximum. They all depend on the input selector.

Note that the Input selector also has a selection option called *Sum*. In this case the meters are fed from the sum of all input channels. This is the default setting.

Also note that the rate meters have a fixed gate time of 100 ms and correspondingly limited accuracy, notably at low rates. They are only a means of quick diagnostics and should not be taken for actual measurements.

7.2. Menus

7.2.1. File Menu

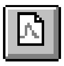
The *File* menu offers the following commands:

New	Clears all histogram data and restores default settings.
Open	Opens an existing histogram file.
Save	Saves a histogram file.
Save As…	Saves an opened histogram file to a specified file name.
Print	Prints the currently displayed histogram.
Page Setup	Allows modifying the page layout for printing.
Print Preview	Displays the layout as it would appear printed.
Print Setup	Selects a printer and printer connection.
1…4 <Recent Filename>	Opens one of the four last recently opened files
Exit	Exits HydraHarp 500 software.

New command

Use this command to create a blank histogram with the last default settings. You can open an existing histogram file with the *Open* command.

Shortcuts

Toolbar: 

Keys: `<Ctrl>+N`
`<Alt>+F N`

Open command

Use this command to open an existing histogram file. All control panel settings will also be restored. You can also open HydraHarp 500 histogram files (`*.phu`) by double clicking them or dragging them onto the Hydra-Harp 500 icon. You can revert to a blank histogram and default control panel settings with the New command. You can also select other file types for loading as histogram data, notably T3 mode files. Loading such files as histogram data is for diagnostics only and cannot use their full information content.

Shortcuts

Toolbar: 

Keys: `<Ctrl>+O`
`<Alt>+F O`

File Open dialog box

The following options allow you to specify which file to open:

Look in: Select (i.e. browse into) the directory in which the file that you want to open resides.

Main box In this box you see the content of the chosen directory, filtered by *Files of type.*

File name: Type or select the filename you want to open. This box lists files with the extension you select in the *Files of type* box.

Files of type Select the type of file you want to open. The default HydraHarp 500 extension is `.phu`

Save command

Use this command to save the current histogram data to a file with current name and directory. When you save a histogram for the first time, the software displays the *Save As…* dialog box so you can name your file. This command is unavailable if a measurement is running. If you want to change the name and directory of an existing file before you save it, choose the Save As command.

Shortcuts

Toolbar: 

Keys: <Ctrl>+S
<Alt>+F S

Save As… command

Use this command to (re–)name the current histogram data and save. The software displays the Save As… dialog box so you can name your file. To save a file with its existing name and directory, use the Save command.

File Save As… dialog box

The following fields allow you to specify the name and location of the file you are about to save:

Save in: Select (i.e. browse into) the directory where you want to store the file.

Main box In this box you see the content of the chosen directory, filtered by *Save as type.*

File name: Type a new filename to save a histogram with a different name. The software automatically adds the extension you specify in the *Save as type:* box.

Save as type: Selects a filter on the directory chosen in the *Save in:* box. Only files that pass the filter are shown in the dialogs main box. Additionally the software adds the extension associated with the chosen filter to a given file name if it was given without extension.

Print command

Use this command to print the currently displayed histogram curves. This command presents a Print dialog box, where you may specify the number of copies, the destination printer, the paper orientation and other printer setup options.

Shortcuts

Toolbar: 

Keys: <Ctrl>+P
<Alt>+F P

Page Setup command

Use this command to change the print layout. This command presents the page setup dialog box, where you may select or deselect various items to appear in the print. You can use print preview to check the resulting layout.

Shortcut

Keys: <Alt>+F U

Print Preview command

Use this command to display the layout as it would appear when printed. The main window will be replaced with a print preview window in which one or two pages will be displayed in their printed format. The print preview toolbar offers you options to view either one or two pages at a time; move back and forth through the document; zoom in and out of pages; and initiate a print job.

Shortcut

Keys: <Alt>+F V

Print Setup command

Use this command to select a printer and a printer connection. This command presents a Print Setup dialog box, where you specify the printer and its connection.

Shortcut

Keys: <Alt>+F R

1, 2, 3, 4 command (most recently used files)

Use the numbers and filenames listed at the bottom of the *File* menu to open the last four files you closed. Choose the number that corresponds with the file you want to open.

Shortcuts

Keys: <Alt>+F 1
<Alt>+F 2
<Alt>+F 3
<Alt>+F 4

Exit command

Use this command to end your HydraHarp 500 session. Save your data before exiting.

Shortcuts

Mouse: Click the close button on the title bar.
Keys: <Alt>+<F4>
<Alt>+F X

7.2.2. Edit Menu

The *Edit* menu offers only one command:

Copy	Copies the displayed curves.

Copy command

Copies the currently displayed curves in ASCII format to the clipboard. This can be used to export histogram data to spreadsheet or data analysis software, e.g., the "EasyTau 2" Software. Note that only the currently selected curves within the current display limits are copied. If the curves in the display are of different resolution then all 65536 time bins will be copied. Copying data to the clipboard replaces the contents previously stored there.

The exported data is plain <Tab> separated ASCII and can be read by most commonly used spreadsheet programs or data analysis software.

This command is unavailable while a measurement is running.

Shortcuts

Keys: <Ctrl>+C
<Alt>+E C

7.2.3. View Menu

The View menu offers the following commands:

Toolbar	Shows or hides the toolbar.
Status Bar	Shows or hides the status bar.
Axis Panel	Shows the axis settings panel.
Control Panel	Shows the control panel.
Trace Mapping	Shows the trace mapping dialog.

Toolbar command

Use this command to display and hide the Toolbar, which includes buttons for the most common commands, such as *File Open*. A check mark appears next to the menu item when the Toolbar is displayed.

Shortcut

Keys: `<Alt>+V T`

Status Bar command

Use this command to display and hide the status bar, which describes the action to be executed by the selected menu item or depressed toolbar button, the current measurement activity and keyboard latch state. A check mark appears next to the menu item when the status bar is displayed. The status bar is displayed at the bottom of the HydraHarp 500 main window.

Shortcut

Keys: `<Alt>+V S`

Axis Panel command

Use this command to display the axis settings panel. The result is the same as the axis panel button in the toolbar or double clicking on the histogram axes. The axis settings panel can be closed by clicking its close button.

Shortcut

Keys: `<Alt>+V A`

Control Panel command

Use this command to display the control panel. This command has the same effect as the control panel button in the toolbar. The control panel can be closed by clicking its close button.

Shortcut

Keys: `<Alt>+V C`

Trace Mapping command

Use this command to display the trace mapping dialog. This command has the same effect as the corresponding button in the toolbar. The trace mapping dialog can be closed by clicking its close button.

Shortcut

Keys: `<Alt>+V M`

7.2.4. Help Menu

The *Help* menu offers the following commands, which provide you assistance with this application:

Help Topics	Offers you the contents list of topics on which you can get help.
Help Index	Offers you an index to topics on which you can get help.
Help Search	Offers you a means of full text search on all help topics.
Activate Context Help	Switches the cursor mode to “point to item” for context help.
Check for Updates	Checks for the availability of a newer software version.
Visit Website	Opens www.picoquant.com in your browser.
Request Support	Provides support details and access to the support website.
About HydraHarp 500 …	Displays version information of the HydraHarp 500 software.

Note: Online help (context help) on most functions, dialogs, control items etc. is available via the F1 key.

Help Topics command

This command opens a browser offering a tree view like list of contents, and a help page. Browse through the contents with your mouse or the cursor keys. `<Enter>` opens / closes chapters, `<Cur Up>` / `<Cur Down>` turns the pages. The help page to the right of the browser will always be held up–to–date.

Shortcuts

Keys: `<F1>` (if online help was last recently opened in topics mode)
`<Alt>+H T`

Help Index command

This command opens a browser offering an index list of keywords and a help page. Browse through the contents with your mouse or the cursor keys. `<Enter>` opens the current item, `<Cur Up>` / `<Cur Down>` navigates through the index.

Shortcuts

Keys: `<F1>` (if online help was last recently opened in index mode)
`<Alt>+H I`

Help Search command

This command opens a browser offering full text search on words or word groups over the whole online help volume. You can enter more specific search pattern, using the operators "*AND*", "*OR*", "*NEAR*" and "*NOT*".

Shortcuts

Keys: `<F1>` (if online help was last recently opened in search mode)
`<Alt>+H S`

Activate Context Help

Changes the mouse cursor mode to context help mode. You can obtain context sensitive help for a visual element in the GUI by simply clicking on it.

Shortcuts

Keys: `<Shift>+<F1>`
`<Alt>+H C`

Check for Updates command

Use this command to check if newer software is available. This requires internet access. If a newer version is available it can immediately be downloaded. Note that this does not perform any installation. You still need to unzip and install the downloaded software. Do this only after reading the release notes.

Shortcut

Keys: `<Alt>+H U`

Visit Website command

Use this command to open www.picoquant.com in your browser. Of course this requires internet access.

Shortcut

Keys: `<Alt>+H W`

Request Support command

Use this command to open a form with important support details of your system and direct access to the support page on the web. The latter requires internet access. If you cannot access the web directly, please copy the support details and send them later by email to support@picoquant.com together with a precise description of the problem. If your setup is not working at all, provide at least the serial number of your HydraHarp 500 and a precise description of the installation environment and the observed issues, including the exact wording of any occurring error messages.

Shortcut

Keys: `<Alt>+H S`

About HydraHarp 500 ... command

Use this command to display the copyright notice and version number of your HydraHarp 500 software and hardware, if installed. It also provides access to the PicoQuant Web site and software updates.

Shortcut

Keys: `<Alt>+H A`

7.3. Toolbar

Toolbar buttons of special interest are explained in some more detail here:

Context Help

When you choose the *Context Help* button, the mouse pointer will change to an arrow with question mark. If you then click somewhere in the HydraHarp 500 window, such as on another toolbar button, the help topic associated with it will be shown.

Axis Panel

Clicking this button opens the *Axis Panel*. This panel provides controls for the axis limits and lin / log display switching. Double-clicking the axes opens the same dialog. The dialog is non-modal, so it can remain permanently open. The panel will remember its last position when closed. When it is opened the next time it will be at that screen position again.

Data Cursor

Opens the data cursor dialog box. You can use this dialog to mark and retrieve the data values of individual histogram bins. A pair of crosshairs will be provided for marking data points with the mouse or keyboard. When the data cursor dialog is activated, the crosshairs will appear in the histogram display area. By clicking on or near a data point, the current crosshair (black) will snap to that point. At the same time a second crosshair (grey) will jump to the previously marked point. The data cursor dialog then shows the current count and time values for these points as well as the corresponding differences (deltas). The data cursor always applies to one "active" curve only. You can select this active curve at the top of the dialog where it is shown with its corresponding curve color. While the grey crosshair normally jumps to the previous data point, you can modify this behavior by holding down the SHIFT key while clicking on curve points. The grey crosshair will then remain at its previous position. This allows finding a particular point while keeping the other one fixed.

Another advanced mode of operation of the data cursor is possible via the LEFT and RIGHT arrows of the keyboard. Pressing these keys will direct the black cursor to the next left or right point in the curve. Again, the behavior of the marker for the previous point (grey cross) can be controlled via the SHIFT key.

Control Panel

Launches the HydraHarp 500 *Control Panel*. If the control panel is already open, subsequent clicking of this button will just make the control panel the active window. The dialog is non-modal, so it can remain permanently open. The panel will remember its last position when closed. When it is opened the next time it will be at that screen position again, even in the next HydraHarp 500 session.

Trace Map

Launches the *Trace Mapping* dialog. The HydraHarp 500 software can measure histograms in up to 512 memory blocks. Out of these, up to 16 curves can be displayed. Note that this limitation applies even to hardware with more input channels, simply because displaying more traces at a time would result in massive clutter on the screen. The trace map dialog is used to select the curves to display. Tick the individual "S*how*" boxes to display a curve. Select the number of the memory block you wish to map the individual display curve to. Choose the active memory block you wish to use for the next measurement in the HydraHarp 500 control panel.

TTTR Mode

Opens the TTTR mode dialog box. Use this dialog to enter the acquisition time and the destination file for a TTTR mode run. Make sure all other measurement parameters have been set and tested in interactive mode before entering TTTR mode.

TTTR Real–Time Correlator

Opens the TTTR mode real-time correlator dialog box. Use this dialog for FCS preview during a TTTR mode run. Make sure all other measurement parameters have been set and tested in interactive mode before entering TTTR correlator mode. Changing to this mode takes a few seconds for reconfiguration of the hardware. See also section 6.3.7.

SEQ Mode Configuration

In order to measure sequences of histograms the HydraHarp 500 software provides a SEQ measurement mode, that allows to call custom code to control custom hardware. This could be a monochromator controlled via stepper motors for automated collection of spectrally resolved lifetime histograms. Clicking this button will launch the dialog for SEQ mode setup. There you can set parameters such as the sequence length, a pause time and the custom code to be called. Demo code fo an executable in C, a batch file and a Python script can be found in the subfolder *SEQdemo* under the same path where the HydraHarp 500 was installed.

General Settings

Opens the software settings dialog box. Use this dialog to change standard settings of the HydraHarp 500 software. Notably these are: *Display rate* (0.1 to 1s), *Draw mode* (lines, stairs), *Grid* check box, *Prompt overwrite* (warning before overwriting existing data), selective disabling of *Warnings* and *TTTR Marker Settings.* The control connector of the HydraHarp 500 provides four LVTTL inputs for synchronization signals. The markers can be enabled or disabled as well as recorded at either the rising or falling edge of the corresponding LVTTL signal. The active edges can be chosen here. There is also a programmable hold-off time to suppress glitches on the marker signals. The dialog also allows selective disabling of warnings that you do not wish to receive. All settings will be kept in the Windows registry and will be retrieved at the next program start. They are stored individually for each Windows user profile.

White Rabbit Settings

Opens the White Rabbit dialog. Use this dialog to configure and establish a White Rabbit link for clock synchronization. White Rabbit allows time transfer and synchronization over fiber based Ethernet with sub-ns accuracy (for more information see https://en.wikipedia.org/wiki/The_White_Rabbit_Project). Note, however, that White Rabbit support is still basic, to some extent simply because White Rabbit as such is still evolving. Currently the HydraHarp 500 supports only basic White Rabbit functionality, notably clock synchronization and time transfer between two HydraHarp 500, where one device is White Rabbit master and the other is White Rabbit slave. Similarly some other White Rabbit device running as a master can act as the clock source of the HydraHarp 500, or the HydraHarp 500 can act as a master and clock source for some other White Rabbit device. Not all of the possible variants have yet been fully tested. White Rabbit support is expected to be improved with future software and firmware updates. Please contact us with your requirements if you are interested in improvements or extensions of this product feature. Also consider our demo code for using White Rabbit in custom software built with the HydraHarp 500 programming library. One particularly useful feature it shows is how one HydraHarp 500 can remote-start another, in order to perform a joint measurement with time alignment.

7.4. Control Panel

The control panel consists of several sections containing edit boxes and other controls for related parameters. These are described in the following subsections. Note that the settings of the control panel as well as the position of the control panel on the screen will be stored in the registry and retrieved at the next program start. When you load a HydraHarp 500 data file the settings of the control panel will change according to the settings in that file. This can be used as a means of reverting to a standard setting for the given experiment.

7.4.1. Sync–Input / Trigger Out

The settings in this group configure the behavior of the Sync input and the Trigger output. See section 6.1 "Setting Up the Input Channels" to get started.

Sync Trigger Mode radio buttons **Edge Tr.** and **CFD**

Here the trigger mode (edge trigger or constant fraction discriminator) for the Sync input can be set. Hardware model dependent the Sync input may not have a CFD. The input will then default to edge trigger and the radio buttons will remain disabled (grayed out).

Sync Trigger Edge edit box and spin control (when in edge trigger mode)

Here the trigger edge (rising/falling) for the Sync input can be set. The value 0 stands for falling, 1 stands for rising edge. Type the value as an integer in the edit box and press `<Enter>` or click on *Apply*. Alternatively, use the spin control (next to it) to increment / decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

Sync Trigger Level edit box and spin control (when in edge trigger mode)

Here the trigger level for the Sync input can be set. Units are millivolts (mV), the permitted range is -1200 to 1200. Type the value as an integer in the edit box and press `<Enter>` or click on *Apply*. Alternatively, use the spin control (next to it) to increment / decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

Sync CFD Zero Cross edit box and spin control (when in CFD mode)

Here the zero cross voltage for the Sync CFD can be set. Units are millivolts (mV), the permitted range is -100 to 0. Type the value as an integer in the edit box and press `<Enter>` or click on *Apply*. Alternatively, use the spin control (next to it) to increment / decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

Sync CFD Level edit box and spin control (when in CFD mode)

Here the discriminator level for the Sync input can be set. Units are millivolts (mV), the permitted range is -1200 to 0. Type the value as an integer in the edit box and press `<Enter>` or click on *Apply*. Alternatively, use the spin control (next to it) to increment / decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

Sync Offset edit box and spin control

Shifts the relative timing of Sync and photon events and is designed to compensate optical and electrical delays (due to differences in cable lengths and optical paths), e.g., in order to shift your fluorescence decay within the sync frame so that it is not truncated. This feature completely eliminates the need for adjustable delay boxes.

Units are picoseconds (ps), the permitted range is from −99,999 to +99,999 ps. Type the value as an integer in the edit box and press `<Enter>` or click on *Apply*. The offset value can be entered in steps of picoseconds but note that the actual step is one time bin according to the instrument's base resolution (1 ps). Alternatively, use the spin control next to the edit box to increment ∕ decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

Sync Tdead edit box and spin control

Tdead is a programmable dead time for suppression of afterpulsing artefacts of some detectors. On the sync input it is rarely needed. Leave it at 0 if you work with a sync signal from a laser.

Units are nanoseconds (ns), the permitted range is from 0 to 160 ns. Note that Tdead = 0 does not really mean zero deadtime, instead, the input will then operate at its shortest (native) dead time of about 680 ps. When a dead time >0 is set then the timing hardware will suppress events falling into the dead time. This is only a suppression of further processing and does not prevent the TDC to go into another (native) dead time. Note that the CFD (when present and used) introduces an extended deadtime of about 6.5 ns anyhow.

Type the value as an integer in the edit box and press `<Enter>` or click on *Apply.* The value can be entered in steps of nanoseconds but note that the actual step is approximated only to the instrument's base resolution (1 ps). Alternatively, use the spin control next to the edit box to increment ⁄ decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

Sync Divider edit box and spin control

Here the programmable divider of the Sync input can be set. This allows to reduce the sync input rate so that the period is at least as long as the dead time. This is required for fast sync sources (≥85 MHz). Internal logic determines the sync period and re–calculates the sync signals that were divided out. It should be noted that this only works with stable sync sources that provide a constant pulse-to-pulse period. All fast laser sources known today meet this requirement within an error of a few picoseconds. Set the divider only as large as necessary. With random and low rate signals (< 1 MHz), the divider must be set to 'None'.

Trigger Output Period edit box and spin control

The trigger output can be used to trigger a light source. Here the period of this output can be set. The unit is microseconds. The allowed range is 0.1 µs to 1.6 s.

Trigger Output Force On check box

Here the trigger output can be switched on regardless of whether a measurement is running.

Trigger Output Auto On check box

Here the trigger output can be set to turn on only when a measurement is running. This can reduce bleaching.

7.4.2. Inputs 1..8, 9..16

The settings in these dialog tabs configure the behavior of the input channels. See section 6.1 "Setting Up the Input Channels" to get started. The HydraHarp 500 can have up to 16 input channels. Dependent on how many channels the device has, one or more tabs with control elements for the channels will be shown. The specific controls for each channel are described below.

Trigger Mode radio buttons **Edge Tr.** and **CFD**

Here the trigger mode (edge trigger or constant fraction discriminator) for the input can be set. Hardware model dependent the Sync input may not have a CFD. The input will then default to edge trigger and the radio buttons will remain disabled (grayed out).

Trigger Edge edit box and spin control (when in edge trigger mode)

Here the trigger edge for the input channel can be set. The value 0 stands for falling, 1 stands for rising edge. Type the value as an integer in the edit box and press `<Enter>` or click on *Apply.* Alternatively, use the spin control (next to it) to increment / decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

Trigger Level edit box and spin control (when in edge trigger mode)

Here the trigger level for the respective input channel can be set. Units are millivolts (mV), the permitted range is -1200 to 1200. Type the value as an integer in the edit box and press `<Enter>` or click on *Apply*. Alternatively, use the spin control (next to it) to increment / decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

CFD Zero Cross edit box and spin control (when in CFD mode)

Here the zero cross voltage for the input CFD can be set. Units are millivolts (mV), the permitted range is -100 to 0. Type the value as an integer in the edit box and press `<Enter>` or click on *Apply*. Alternatively, use the spin control (next to it) to increment / decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

CFD Level edit box and spin control (when in CFD mode)

Here the discriminator level for the input CFD can be set. Units are millivolts (mV), the permitted range is -1200 to 0. Type the value as an integer in the edit box and press `<Enter>` or click on *Apply*. Alternatively, use the spin control (next to it) to increment / decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

Offset edit box and spin control

Shifts the relative timing of photon events and is designed to compensate optical and electrical delays (due to differences in cable lengths and optical paths), e.g., in order to shift your fluorescence decay within the sync frame so that it is not truncated. This feature completely eliminates the need for adjustable delay boxes.

Units are picoseconds (ps), the permitted range is from −99,999 to +99,999 ps. Type the value as an integer in the edit box and press `<Enter>` or click on *Apply*. The offset value can be entered in steps of picoseconds. Alternatively, use the spin control next to the edit box to increment ⁄ decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

Tdead edit box and spin control

Tdead is a programmable dead time for suppression of afterpulsing artefacts of some SPAD detectors. Otherwise it is rarely needed. Leave it at 0 unless you have good reasons.

Units are nanoseconds (ns), the permitted range is from 0 to 160 ns. Note that Tdead = 0 does not really mean zero dead time, instead, the input will then operate at its shortest (native) dead time of about 680 ps. When a dead time >0 is set then the timing hardware will suppress events falling into the dead time. This is only a suppression of further processing and does not prevent the TDC to go into another (native) dead time. Note that the CFD (when present and used) introduces an extended deadtime of about 6.5 ns anyhow.

Type the value as an integer in the edit box and press `<Enter>` or click on *Apply*. The value can be entered in steps of nanoseconds but note that the actual step is approximated only to the instrument's base resolution (1 ps). Alternatively, use the spin control next to the edit box to increment ⁄ decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

7.4.3. Acquisition

Resolution edit box and spin control

Use this set of input controls to specify the time resolution. Units are picoseconds (ps). Possible choices are the device's base resolution and successive multiples by two. Type the desired resolution value as an integer in the edit box and press `<Enter>` or click on *Apply*. Alternatively, use the spin control (next to it) to increment / decrement the current value. In this case, changes take effect immediately without pressing `<Enter>`. In case of entering values other than valid resolutions, the next suitable resolution step is chosen automatically.

There are always 2^{16} = 65536 time bins in one histogram. With the chosen resolution, the respective time range covered is 65536 * Resolution. The choice of range must be a compromise between resolution and time span covered. The smallest range offers the best resolution and the shortest span (vice versa for the largest range). For high sync rates the highest resolution is usually most appropriate, since the smallest range still covers the full sync period. For lower sync rates the histogram range may be too small to cover the full sync period. The decay curve region of interest may therefore lie outside the acquisition window of 65536 time bins. Apart from switching to a lower resolution it is possible to shift the acquisition window relative to the sync frame by means of the offset. Note that working with very long time spans (low sync rates) at high count rates also requires long acquisition times to minimize noticeable “steps” in the acquired histograms.

Acquisition Time edit box and spin control

Set the desired measurement time here. Units are seconds (s). The permitted range is 0.001 to 360,000 s in steps of 0.001 s. Type the desired value in the edit box and press `<Enter>` or click on *Apply*. Alternatively, use the spin control (next to it) to increment / decrement the current value. In this case, changes take effect immediately without pressing `<Enter>`. The default increment / decrement per mouse click is logarithmic.

Offset edit box and spin control

For high sync rates the highest resolution is usually most appropriate, since the smallest range still covers the full sync period. The offset should in these cases always be set to 0.

For lower sync rates the histogram range may be too small to cover the full sync period. The measurement region of interest may therefore lie outside the acquisition window of 65536 histogram bins. Apart from switching to a lower resolution it is possible to shift the acquisition window relative to the sync frame by means of setting the offset > 0.

Use this set of input controls to specify the desired offset. Units are nanoseconds (ns). Possible offsets are in the range from 0 to 100,000,000 ns. Type the value as an integer in the edit box and press `<Enter>` or click on *Apply*. The offset value can be entered in steps of nanoseconds but will internally be rounded to the nearest multiple of the device’s base resolution. Alternatively, use the spin control next to the edit box to increment ⁄ decrement the current value. In this case changes take effect immediately without pressing `<Enter>`.

Internally the offset is subtracted from each start–stop measurement before it is used to address the histogram bin to be incremented. Therefore, increasing the offset means shifting the signal on the screen to the left, towards earlier times.

Note that the offset has no effect on the relative timing of laser pulses and photon events. It merely shifts the region of interest where data is to be collected. The relative timing of laser pulses and photon events can only be controlled by means of cable delays and the input offsets described in sections 7.4.1 and 7.4.2. The latter may be necessary in order to shift your fluorescence decay within the sync frame so that it is not truncated.

Trc./Block trace color indicator, edit box and spin control

The HydraHarp 500 software can measure and store histograms in up to 512 memory blocks. Out of these, up to 16 curves can be displayed and one "active block" can be used for measurements. Choose the active memory block you wish to use for the next measurement here. The trace mapping dialog is used to select the curves for display. Make sure the curve you are using is switched on (*shown*). You can open this dialog directly by clicking on the trace color indicator. The color indicator shows the trace color the chosen block is currently mapped to. If it is a solid square, the curve is mapped and shown. If it is mapped but not shown, the indicator shows a small striped square. If the curve is not even mapped for display the indicator remains white. Make sure not to overwrite existing data in a block that was used before. In order to warn you that a trace is used, the heading of the Trc/Block selector will turn red.

Restart check box

If this box is ticked (checked) the measurement will automatically restart after the acquisition time has elapsed. Toggle the current setting with a mouse click. The setting is without effect in SEQ mode.

Stop on Overflow check box

If this box is ticked (checked) the measurement will stop on overflow of any histogram bin. The overflow limit is set by the "*Stop At*" parameter. Toggle the current setting with a mouse click.

Mode drop-down selection box

Here you can select between three different acquisition modes:

Oscilloscope Mode:

In oscilloscope mode the acquisition and display, once started, repeats at intervals given by the current acquisition time setting. The histogram accumulation always starts from scratch. This is useful for monitoring fast changes during optical alignment etc. Note that in this mode you may see nothing for a long time if you set the acquisition time to a high value. Typically one would not set an acquisition time of more than 1 second in this mode.

Integration Mode:

As opposed to oscilloscope mode, the histogram acquisition in integration mode is not reset to zero with each display refresh. The histogram continues to grow while the display is updated every 0.1 to 1 seconds. The update rate is determined by the refresh rate value selected in the *General Settings Dialog*. Acquisition can be manually started and stopped, additionally the option "*Stop on Overflow*" will stop the acquisition when the maximum count of any histogram bin overflows. The option "*Restart*" will cause the acquisition to start again after the acquisition time has elapsed. This is similar to oscilloscope mode but allows watching the histogram grow if the acquisition time is fairly long.

Sequence Mode:

Selects the measurement mode SEQ for collection of a whole sequence of histograms with an optional call to custom code for hardware control before each measurement. See section 6.4.

Stop At edit box and spin control

Here the stop count level for histogramming can be set. The setting is only meaningful if the check box "*Stop Ofl*" is active. Type the value as an integer in the edit box and press `<Enter>` or click on *Apply*. Alternatively, use the spin control (next to it) to increment / decrement the current value. In this case changes take effect immediately without pressing `<Enter>`. The minimum is 0. If the entered value exceeds the allowed maximum of 4,294,967,295, an error message appears. When using the spin control, exceeding the maximum is not possible. Instead of incrementing the value further, the system beeps as a warning. Note that in case of histogramming on multiple channels the stop will occur for all channels when the prescribed stop count is reached on any channel.

7.5. Axis Panel

The controls of the *Axis Panel* are used to customize the main window's curve display. All mouse actions on the spin controls on this dialog result in instant modification of the display. If you choose to alter the values in the edit fields by keyboard, you have to finalize your changes by pressing `<Enter>` or clicking the *Apply* button.

The controls are grouped as follows:

7.5.1. Time Axis Group

Note that the time axis labeling is precise in terms of relative information only. At offset=0 the absolute time difference between the inputs is close to the shown times but may be off by some tens of picoseconds. Offsets >0 change the placement of the acquisition window and the true time differences are then off accordingly.

Minimum edit box and spin control

Here the starting value of the displayed time axis can be set. Units are nanoseconds (ns). The minimum is 0. If the entered value exceeds the current time axis maximum, an error message appears. When using the spin control exceeding the current time axis maximum is not possible. Instead of incrementing the minimum value further, the system beeps as a warning. Keep in mind the notes on time axis interpretation given in the paragraph *Time Axis Group* above. Also note that the increment upon using the spin buttons grows with the current value. This permits fast changes across the entire range.

Maximum edit box and spin control

Here the end value of the displayed time axis can be set. Units are nanoseconds (ns). If the entered value becomes smaller than the current time axis minimum, an error message appears. When using the spin control, violating the current time axis minimum is automatically prevented. Instead of decrementing the minimum value further, the system beeps a warning. Keep in mind the notes on time axis interpretation given in the paragraph *Time Axis Group* above. Also note that the increment upon using the spin buttons grows with the current value. This permits fast changes across the entire range.

The time axis settings will determine the number of time bins exported via clipboard copy / paste. Only the bins within the displayed time range will be used.

7.5.2. Count Axis Group

Minimum edit box and spin control

Here the starting value of the displayed count axis can be set. Units are counts. If the minimum is set to 0 in logarithmic mode, the resulting minimum as displayed is 10^{-1}. If the entered value exceeds the current count axis maximum, an error message appears. When using the spin control, exceeding the current count axis maximum is automatically prevented. Instead of incrementing the minimum value further, the system beeps a warning. Note that the increment upon using the spin buttons grows with the current value. This permits fast changes across the entire range.

Maximum edit box and spin control

Here the upper end of the displayed count axis can be set. Units are counts. If the entered value becomes smaller than the current time axis minimum, an error message appears. When using the spin control, violating the current count axis minimum is prevented automatically. Instead of decrementing the minimum value further, the system beeps a warning. Note that the increment upon using the spin buttons grows with the current value. This permits fast changes across the entire range.

Lin / Log radio buttons

These radio buttons change the scaling of the count axis from linear to logarithmic and vice versa. Despite the fact that in logarithmic scale the display will always show whole powers of ten, the change of the scaling is applied without modifying the minimum / maximum range.

7.6. Trace Mapping Dialog

The HydraHarp 500 software can measure and store histograms in up to 512 memory blocks. Out of these, up to 16 curves can be displayed at the same time. The trace mapping dialog is used to select the curves to display. Tick the individual boxes '*Show*' to display a curve. Select the number of the memory block you wish to map the individual display curve to. The Trace Mapping Dialog also provides some statistics on each curve. These items are:

FWHM	The Full Width Half Maximum of the curve peak (usually for IRF traces)
Max Count	The count in the highest point of the curve
At Time	The time corresponding to the histogram bin where *Max Count* occurred
Resolution	The time bin width of the curve

Furthermore there are several buttons:

Details	Can be clicked to see more curve information
All	Can be clicked to mark all traces as shown
None	Can be clicked to mark all traces as not shown
0..15	Can be clicked to set the default mapping of trace 0..15 to block 0..15
16..31	Can be clicked to set the mapping of trace 0..15 to block 16..31
32..47	Can be clicked to set the mapping of trace 0..15 to block 32..47
48..63	Can be clicked to set the mapping of trace 0..15 to block 48..63
Clear (trash can)	Can be clicked to delete the contents of individual blocks

The trace mapping dialog can be launched from the corresponding button on the toolbar as well as through the trace color indicator on the control panel.

7.7. General Settings Dialog

Use this dialog to change standard settings of the HydraHarp 500 software. Notably these are: *Prompt overwrite* (warning before overwriting existing data) and *TTTR Marker Settings.* The control connector of the HydraHarp 500 provides four LVTTL inputs for synchronization signals. The markers can be enabled or disabled and recorded at either the rising or falling edge of the corresponding LVTTL signal. The active edges can be chosen here. All settings will be kept in the Windows registry and will be retrieved at the next program start. They are stored on a per user basis. Note that the retrieved settings may become pitfalls when they are set for to values in not in agreement with the measurement you are intending to run.

The dialog's controls are grouped as follows:

Display Group

Display Rate /s	Sets the time interval between display updates (from 0.1 to 1s).
Draw mode	Switches between different curve draw–modes (Lines / Stairs).
Grid	Checked: Shows a light grey grid on the curve display. Unchecked: The axes of the curve display are marked with ticks

File Saving Group

Prompt overwrite	Activate this check box if you wish to be prompted before saving data over existing data in a file (recommended).

Warnings Group

Here you can selectively enable / disable individual warnings. Activate the check box for each warning you wish to receive. See section 9.1 for details.

Input Hysteresis Group

Here it is possible to set a hysteresis of the input comparators larger than the default value. It applies to all inputs simultaneously, including the sync input. There are only two choices: the default value of about 3 mV and the large hysteresis value of about 35 mV. The larger hysteresis may in some cases help to suppress noise artefacts on the input signals. Consider this only a last rescue when it is impossible to eliminate the noise at its origin.

Note that this setting changes upon loading an existing measurement file according to what was set in this file.

TTTR Marker Settings Group

Here you can make settings for the external marker signals used in TTTR mode (T2 and T3 mode). The settings for the different markers are independent from each other. The following table applies to each of the four markers:

enable	Check this if you want the marker to be included into the TTTR data stream
rising / falling	Radio buttons to select the active edge of the maker signal
Holdoff time	Edit box for marker holdoff time in ns (0..25500 ns)

The holdoff time can be used to deal with noisy marker signals and reflections on the marker lines. Once a transition on the selected marker edge was detected the holdoff logic will then suppress any further detection within the prescribed holdoff time. The setting applies to all markers signals. Consider this only a last rescue when it is impossible to eliminate the noise at its origin. Note that the internal resolution of this setting is 100 ns, the entered numbers will be rounded down to the nearest multiple of 100 ns.

Reference Clock Source Group

Here you can select the HydraHarp 500 ‘s clock source. Normally it runs on its own internal crystal clock, which stems from a very accurate oven controlled crystal oscillator (OCXO) with very low temperature drift (± 10 ppb). In cases where clock synchronization with other devices is required it is possible to use an external source, e.g., another HydraHarp 500, a GPS/GLONASS receiver, or an atomic clock. The latter are of interest when a more accurate clock is required. In each case the clock source must deliver an industry standard 10 MHz clock signal. When remote synchronization is required the best option is White Rabbit. In this case the White Rabbit link must have been established before switching to White Rabbit as a clock source. Note that any change of the clock source requires re-initialization of the HydraHarp 500 . Also note that any disconnection from or interruption of the external clock will cause the HydraHarp 500 to fail and report a system error. To recover from this error state the device must be re-initialized after the clock source is stable again.

7.8. White Rabbit Dialog

This dialog allows configuration and control of the HydraHarp 500 ‘s built-in White Rabbit PTP Core (WRPC). This core runs autonomously in the HydraHarp 500 ‘s FPGA and is not normally required to use the HydraHarp 500 ’s regular TCSPC and time tagging functionality. However, once a White Rabbit connection has been established it is possible to switch the HydraHarp 500 ’s clock source over to White Rabbit. Time tagging can then be performed synchronous to the White Rabbit clock and remote device(s) synchronized to the same timing reference.

For general information on White Rabbit please refer to the website
https://www.ohwr.org/projects/white-rabbit.

For information on the White Rabbit PTP Core please refer to the page
https://gitlab.com/ohwr/projects/wrpc-sw/wikis/home.

For details on the specific version 4.0 of the core currently used in the HydraHarp 500 please refer to
https://gitlab.com/ohwr/projects/wr-cores/-/wikis/wrpc-release-v40.

7.8.1. Description of Controls

The White Rabbit Dialog provides three groups of controls that are described in the following subsections.

White Rabbit EEProm Data

In order to use White Rabbit in the HydraHarp 500 it is first necessary to provide the WRPC with some configuration data. This configuration data is stored in Electrically Erasable Programmable Read-Only Memory (EEPROM) and the controls in this group serve the purpose of editing, storing, and retrieving it.

The topmost set of controls labeled **MAC** is used to assign a Media Access Control (MAC) address to the HydraHarp 500 ‘s WR network port. The MAC is essentially a string of 6 bytes and must be unique within your network. If you are connecting to a non-private network then it should also be globally unique*. The button `Read` will retrieve the MAC address from EEPROM and place it as a hexadecimal number in the edit field on the left. If there is no MAC address in EEPROM you may want to enter one in the edit field (as a hexadecimal number) and then click the button `Write` in order to store it in EEPROM. The status field on the right will show success or failure information regarding the last read or write activity.

The next set of controls further down, labeled **Init Script** is for configuration of an initialization script that runs when the WRPC starts (i.e. when the HydraHarp 500 is switched on, or on demand, see below). For details of the script syntax please refer to the WRPC documentation. The procedure of reading and writing the script from/to EEPROM is the same as described for the MAC address above.

The next set of controls further down, labeled **SFP DATA** is for configuration of the Small Form-factor Pluggable transceiver (SFP) module(s) being used. The WR socket at the HydraHarp 500 ‘s front panel is a receptacle for an SFP module converting between electrical and optical signals. The SFP modules require calibration and the group of controls described here allows for depositing the calibration data so that the WRPC can use it. The controls and procedures for editing, reading and writing are the same as described for the other two control groups above. The EEPROM provides space for calibration data of up to 4 SFP modules. This corresponds to 4 lines in the edit box on the left. The syntax for each line is

```
SFP-Module-name   Delta-Tx  Delta-Rx   Alpha
```

where the SFP-Module-name is an ASCII string given by the module manufacturer and the following three calibration values are signed decimal numbers. All items are separated by spaces. Not all lines need to be populated. The specific calibration parameters must be obtained by a calibration procedure according to the WRPC manual. If you order SFP modules from PicoQuant they will be shipped as a pair with the following names and calibration values (valid as of August 2025):

```
AXGE-1254-0531    218466    218726     63293704
AXGE-3454-0531    217080    217320    -60056437
```

You can put these two lines directly into the edit field and write them to EEPROM. Should other models be shipped at a later time, the calibration data will be provided with the modules. If you obtain SFP modules from other sources you will have to request the calibration data from the supplier or perform the calibration on your own. For very precise calibration this is recommended anyhow, as the calibration is to some extent dependent on the particular devices and fibers in use.

WR Status

This group of controls provides selected status information of the WRPC. It my be retrieved manually by clicking the `Refresh` button. It may also be retrieved periodically by checking the tick box `Auto`. Note that the status information is retrieved from the WRPC’s console output, which is available only when the WRPC is running and has received the command `gui`.

Start/Stop/Restart

This group of controls provides a drop-down menu for mode selection and buttons for manual control of the WRPC. The usage of these buttons will be explained further down as part of the description of the related WR initialization procedures.

* Official and guaranteed unique MAC addresses can be obtained from IEEE (http://www.ieee.org)

WR Console Output

The WRPC is internally running an embedded software that provides a command console that in principle can be accessed like using a terminal. The current implementation in the HydraHarp 500 firmware and software does not allow full access to this console. However, the group of controls described here provides read access to the console output. If the tick box `Auto-Refresh` is checked the WR console output is periodically refreshed. Note that the WRPC's console output is providing meaningful status information only when the WRPC is running and has received the command `gui` which starts the WRPC monitor.

7.8.2. Setting up White Rabbit Connections

As outlined above, using White Rabbit as a clock source for the HydraHarp 500 requires establishing a WR connection. The following describes the necessary steps to set up such a connection by the example of a point to point WR connection between two HydraHarp 500 devices.

As a first step it is always necessary to switch the WR link on (starting the embedded WRPC software). This is done by clicking the button `Link On` in the WR dialog of both devices. In order to see if this is successful it is advisable to have the automatic status refresh enabled. It may take a few seconds until the link is established. Now you can select the WR mode of one device as master and the other as slave. Then click the button `Set Mode` in each device's WR dialog. Observe the status display and wait until the PTP status switches to MASTER and SLAVE respectively. The console output should now say “Locked Calibrated” in green letters on both sides and on the slave side the servo state should eventually switch to “TRACK_PHASE”. The WR connection is now established and you can inspect the console output, e.g., to check the timing accuracy between the two devices or other connection statistics. For details please refer to the WRPC manual.

After the WR connection is established it is possible to set the current time of the master side as the common time for master and slave. This can be done by clicking the button `Set Time` at the master. Note that this time is obtained from the operating system (as UTC). This is only as accurate w.r.t. true UTC as the accuracy of the master's Windows clock setting. More accurate time can be obtained by using the Grand Master mode, which, however, requires an accurate time source such as a GPS/GLONASS receiver, an atomic clock or similar, connected to the HydraHarp 500 via the 10MHz clock reference and PPS inputs at the back of the housing. Note however, that such functionality is not yet conveniently supported by the HydraHarp 500 software. Serious work in this direction should use the HydraHarp 500 programming library and custom software. In practice it is probably more convenient to use a dedicated commercial grand master in the WR network.

When the WR connection is established and (optionally) the WR time has been set, the WR dialog can be closed and WR can be selected as the reference clock source. Note that the choice of master versus slave in the clock source selection must match the role of master/slave of the WR connection.

For regular use of the WR connection it is rather cumbersome to perform the described connection procedure manually each time. Instead, it is possible to place an appropriate initialization script in master and slave so that it is automatically executed upon power-up. An example for the slave side may look as follows:

```
ptp stop
sfp match
mode slave
ptp start
gui
```

The only difference for the master side is setting `mode master` (or `mode gm` for grandmaster). Please note that the last line `gui` is not strictly necessary in a generic WRPC node. However, due to the HydraHarp 500 's current implementation it is required because the WRPC monitor output is used to retrieve status information for the related programming library functions and the status information provided in the WR dialog. For more details on the script commands please refer to the WRPC documentation.

So far, only a point to point WR connection between two HydraHarp 500 devices was discussed. It is also possible to connect a HydraHarp 500 with some other WR node and across WR network switches. So far this was tested with the “SPEC board” and the WR switches from Safran (formerly Seven Solutions). Please contact us if you wish to operate the HydraHarp 500 with other WR hardware or in a more complex network so that we can advise or consult with the vendor. Furthermore, some advanced uses of WR such as remote experiment control and joint action of multiple devices are not yet fully implemented. We look forward to collaborate with interested users as well as suppliers in order to extend and improve the HydraHarp 500 's WR interoperability and functionality. Please get in touch.

7.9. About HydraHarp… Dialog

This dialog provides version information on the HydraHarp 500 software and hardware, the latter only if it is connected and operational. It can be opened via the help menu or via the toolbar. The button *Request Support* opens a small text viewer window that provides a listing of HydraHarp 500 hardware and software versions that you can copy and paste into your support enquiry. This information is very important for adequate support. Support requests without this information cannot be processed and will be delayed by return questions for this information. If your system is not functional at all, the minimum information you must provide for support is the serial number of your HydraHarp 500 . It can be found at the back of the housing. The dialog also provides buttons for links to relevant web pages and a button for checking for and downloading software updates. Note that downloading a software update does not automatically install it. The downloads contain zip files that must be unpacked and installed manually. Even though updates are typically bringing improvements and bugfixes, please note that it may not always be advisable to blindly install the latest software. This is the case especially when a user or his/her laboratory have developed custom software that relies on a certain feature of the old version that may have changed. Updates may break such compatibility. Therefore, always carefully read the release notes before installing and/or ensure a safe fallback.

7.10. Title and Comment Editor

You can use this dialog to edit the file title / comment. It can be opened via double–click on the title, via the *File* menu or via the *Print Preview* Toolbar. The text you enter here will be stored in the data file. The first line will be displayed as the file title above the histogram display area. The text you can enter here is limited to 4 lines and 255 characters. Upon loading of a data file, the title and comment will also be retrieved. It will also be included in prints, if the corresponding check mark is set in the page setup dialog box available through the *File* menu.

7.11. Print Preview Dialog

Use this Dialog to preview the layout as it would appear when printed. When you choose this command, the main window will be replaced with a print preview window in which one or two pages will be displayed in their printed format. The print preview dialog offers the following options:

Print

Bring up the print dialog box to start a print job.

Next Page

Preview the next printed page.

Prev Page

Preview the previous printed page.

One Page / Two Page

Preview one or two printed pages at a time.

Zoom In

Take a closer look at the printed page.

Zoom Out

Take a larger look at the printed page.

Page Setup

Change the layout of the printed page.

Close

Return from print preview to the editing window.

8. Problems, Tips & Tricks

8.1. Basic Pitfalls

The hardware settings in the HydraHarp 500 software are retrieved to the current settings upon loading an existing measurement file. This is a convenient feature most users like because it easily permits repeating a measurement with identical settings. However, the retrieved settings may become pitfalls when they are not in agreement with the measurement you are intending to run. This may be general settings (see section 7.7) as well as control panel settings (see section 7.4). If measurements behave strangely, always check your active settings in both places.

8.2. PC Performance Issues

The HydraHarp 500 device and its software interface are a complex real–time measurement system requiring appropriate performance both from the host PC and the operating system. This is why a fairly modern CPU and sufficient memory are required, along with a recent USB 3.x (or compatible) host controller. The screen resolution should be at least 800x600. At least a 2 GHz dual core (better quad core) processor, 8 GB of memory and a fast hard disk (preferably SSD) are recommended.

In order to maintain correct interaction between the measurement hardware, the display of histogram curves and user input, the operating system's message passing mechanism is used. It is recommended not to overload the system by running other processes in the background while measuring with the HydraHarp 500 . The PC's own occasional network activity should be no problem but running the machine e.g., as a server for other PCs is not recommended. In principle any kind of background activity is allowed. However, should the system become overloaded, photon events may be lost and measurement times and display rates may become irregular.

You can minimize the running HydraHarp 500 software during measurements without issues. This may be of interest for lengthy measurements in integration mode, where one is only interested in accumulating a certain amount of counts without need for permanent monitoring. However, the things you do in the meantime must not overload the CPU.

The "panel meters" showing the current count rates are not meant to be 100 % accurate. They merely serve as an aid for setting up the system. Some of them may suffer from system overload. Accordingly the values shown in the curve details (trace mapping dialog) are subject to such tolerances.

In TTTR mode, system overload will manifest itself in loss of data and termination of the measurement. Especially the real-time correlator requires significant resources. Here no other heavy load background processes should be allowed to run. The faster the PC, the less these issues will matter. TTTR mode performance (i.e. max. time tagging throughput to disk) may also be improved by using modern solid state disks.

If you are working with other bandwidth demanding USB devices, e.g. USB cameras, consider that the overall USB throughput is limited by the host controller or any hub the devices must share. It is advisable to connect such devices to separate host controllers without sharing hubs. If you install additional USB controller cards you should prefer fast PCI–express models. However, modern mainboards often have multiple USB host controllers, so you may not even need extra controller cards. In order to conveniently find out how many USB controllers you have and which one the individual USB sockets belong to, you can use Microsoft's Universal Serial Bus Viewer (*USBView*). The HydraHarp 500 software setup also installs a USB speed test utility `HH500speedtest.exe` that you can use to measure the USB throughput of the HydraHarp 500 at its present connection. If the connction is good and the PC is sufficiently fast it should show a sustained throughput of about 350 MB/s. This corresponds to a throughput of about 88 Mevents/s in TTTR mode. The tool also records and shows the count of USB errors. It is normal that there are a few hundred phy errors at startup, however, if they increase during the test run the connection is not stable.

8.3. Histogram Artefacts

Disturbing histogram ripple is strongly dependent on the quality of the input signals. Try to deliver the best possible signal quality with clean and reasonably steep active edges and without too much ringing. It is recommended to use detector and sync signals of similar amplitude to minimize cross-talk. If a picosecond pulsed laser from PicoQuant's PDL Series is used, attenuating the sync pulses by 10 dB immediately at the laser driver may reduce histogram ripple (set trigger level accordingly). Always use good quality 50 Ω components and coax cables with proper shielding and correct termination. We recommend cables of type RG223 with double braided and silver plated screen. Check your setup for ground loops. Grounding different system components (PC, detector, detector power supply, diode laser driver, etc.) at different points can induce considerable noise in the ground lines. Because signal return paths may share the same ground lines, this noise is copied into the signal lines and causes increased timing jitter and / or histogram ripple. Network cables and mobile phones may also be sources of noise. Use properly impedance matched power splitters (reflection–free T–pads) if signals must be fed to multiple 50 Ω inputs. Never use ordinary BNC tees. All accessories are available from PicoQuant. PMT detectors should be connected through a suitable high speed preamplifier (available from PicoQuant). MCP–PMT detectors should be connected through an amplifier with slightly higher gain (also available from PicoQuant). TTL–SPAD–detectors (e.g. Perkin–Elmer SPCM–AQR and successors by Excelitas) must be connected through an attenuator to avoid damage. In case of positive going signal such as (attenuated) TTL, the inputs must be used in ETR mode, the CFDs can only handle negative going signals.

An often overlooked problem in fluorescence lifetime measurements with TCSPC is pile–up and dead time related histogram distortion. It becomes noticeable if the detector count rate exceeds ~5 % of the sync rate. This is why high excitation rates are important. The effect is an inherent problem of high resolution TCSPC and not a fault of the hardware. The HydraHarp 500 is less significantly affected by such issues because of its very short dead time. Nevertheless, the safest way to handle dead time related histogram distortions is to maintain count rates < 5 % of the excitation rate. If high throughput is a primary objective and the detector dead time is not limiting, then this rule of thumb may of course be intentionally ignored. This may bring about the issue of pulse-pile-up because individual detector pulses begin to overlap. Pulse pile-up may also be corrected for in data analysis. Indeed, combined with suitable detectors (e.g., PMA Hybrid from PicoQuant) the HydraHarp 500 will allow fluorescence lifetime measurements with count rates as high as 80 MHz, i.e. as fast as the typically highest excitation rate. For further details see the literature given at the end of section 3.4 and the publications on PicoQuant's innovative concept of rapidFLIM.

8.4. Warming Up Period

Observe the warming–up period of at about 20 minutes (depending on ambient temperature) before using the HydraHarp 500 for important measurements. You can use this time for set–up and preliminary measurements. The maximum permissible ambient temperature is 35 °C. Make sure that the cooling air can circulate freely and no other hot instrument is sitting directly under the HydraHarp 500.

8.5. Custom Programming of the HydraHarp 500

A programmer's library (DLL) for custom Windows software development is available to build your own applications e.g., in LabVIEW, Matlab, Python, C/C++, C#, Rust, and Pascal (Delphi/Lazarus). A rich set of demo code is provided for an easy start. If you care about performance, consider using a proper compiled high level programming language such as C/C++, Rust, or Pascal. Scripted languages like Matlab and Python tend to be very slow. There is also a library version for Linux (x86-64 processor architecture only) which is fully compatible with that for Windows so that applications can easily be ported across the two platforms. A relatively advanced high-level API package “snAPI” for Python is also available (Windows only). It readily provides data collection and file writing methods as well as many real-time analysis methods such as intensity and coincidence time traces, FCS and $g^{(2)}$ correlation.

8.6. Software Updates

We constantly improve and update the software for our instruments. This includes updates of the configurable hardware (FPGA). Such updates are important as they may affect reliability and interoperability with other products. The software updates are free of charge, unless major new functionality is added. The latest software is available for download at the PicoQuant web site. Alternatively you can click the button 'Check for Updates' in the Help-About dialog available through the main menu or the toolbar. Note that downloading a software update by means of this button does not automatically install it. Typically the downloads contain zip files that must be unpacked and installed manually. Also note that it may not always be advisable to blindly install the latest software. This is the case especially when a user or his/her team have developed custom software that relies on a certain feature that may have changed. Therefore, always carefully read the release notes before installing updates.

8.7. Support and Bug Reports

The HydraHarp 500 TCSPC system has gone through extensive testing. It builds on over 25 years of experience with several predecessor models and the feedback of hundreds of users. Nevertheless, it is a fairly complex product and some bugs may still be found. In any case we would like to offer you our support if you experience problems with the system. Do not hesitate to contact PicoQuant in case of difficulties with your HydraHarp 500 .

As a first step it is always advisable to study the manual or to press F1 for help. Should you observe errors or bugs caused by the HydraHarp 500 system please try to find a reproducible error situation. Then open *Help - Request Support*. This dialog provides important version information on the HydraHarp 500 software and hardware (the latter only if it is connected and operational). It can be reached via the Help menu or via the toolbar. It provides a small text viewer window with a listing of HydraHarp 500 hardware and software versions that you can copy and paste into your support request. Note that this information is very important for adequate troubleshooting. Support requests without this information cannot be processed and will be delayed by return questions for this information. If your system is not functional at all, the minimum information you must provide for support is the serial number of your HydraHarp 500 . It can be found at the back of the housing. There may be other relevant circumstances, especially other new hardware installed in your PC, so please provide details. You can run *msinfo32* to obtain a listing of your PC configuration and attach the summary file to your error report to www.picoquant.com/contact/support. If you cannot access the web form directly, please contact support@picoquant.com and include the same information. Complete information will help us to help you more quickly. When you submit an error report referring to measurement data you have taken, attach the original data file. If the file is too large for email (>5 MB) please provide access to it on a public file server.

If the device must to be sent in for inspection / repair / upgrade, please request an RMA number before shipping the hardware. Observe precautions against static discharge, moisture, and mechanical damage under all circumstances in handling, packaging, and shipping. Use original or equally protective packaging material.

9. Appendix

9.1. Warnings

When the HydraHarp 500 software is running with functional hardware it continuously collects information about the input signals and the current acquisition settings. If these settings along with the input rates indicate possible errors, the software will indicate this by showing a warning icon.

While the software is running in interactive histogramming mode the warning icon is displayed at the bottom of the main window in the status bar. In the case of TTTR mode, it will appear directly in the TTTR mode dialog. The icon can be clicked to display a list of current warnings together with a brief explanation of each warning. Similarly the Warnings (if any) will be stored in the data files and can be inspected via the Curve Details dialog, also after re-loading such a file later.

The warnings are to some extent dependent on the current measurement mode. Not all warnings will occur in all measurement modes. Also, count rate limits triggering a specific warning may be different in the various modes. The following table lists the possible warnings in the three measurement modes and gives some explanation as to their possible cause and consequences.

Warning	Histo Mode	T2 Mode	T3 Mode
WARNING_SYNC_RATE_ZERO No pulses are detected at the sync input. In histogramming and T3 mode this is crucial and the measurement will not work without this signal.	√		√
WARNING_SYNC_RATE_VERY_LOW The detected pulse rate at the sync input is below 100 Hz and cannot be determined accurately. Other warnings may not be reliable under this condition.	√		√
WARNING_SYNC_RATE_TOO_HIGH The pulse rate at the sync input (after the divider) is higher than 82 MHz. This is close to the sustainable front end speed. Sync events may be lost above 85 MHz. T2 mode is normally intended to be used without a fast sync signal and without a divider. If you see this warning in T2 mode you may accidentally have connected a fast laser sync.	√	√	√
WARNING_INPT_RATE_ZERO No counts are detected at any of the input channels. In histogramming and T3 mode these are the photon event channels and the measurement will yield nothing. You might sporadically see this warning if your detector has a very low dark count rate and is blocked e.g. by a shutter. In that case you may want to disable this warning.	√	√	√
WARNING_INPT_RATE_TOO_HIGH The overall pulse rate at the input channels is higher than 85 MHz (USB 3.0 connection) or higher than 9 MHz (USB 2.0 connection). This is close to the throughput limit of the present USB connection. The measurement will likely lead to a FIFO overrun. There are some rare measurement scenarios where this condition is expected and the warning can be disabled. Examples are measurements where the FIFO can absorb all data of interest before it overflows.	√	√	√

Warning	Histo Mode	T2 Mode	T3 Mode
WARNING_INPT_RATE_RATIO This warning is issued in histogramming and T3 mode when the rate at any input channel is higher than 5% of the sync rate. This is the classic pile-up criterion. There are some measurement scenarios where this condition is expected and the warning can be disabled. Examples are antibunching measurements or Rapid-FLIM where pile-up is either tolerated or corrected for during data analysis. One can usually also ignore this warning when the current time bin width is larger than the dead time.	√		√
WARNING_DIVIDER_GREATER_ONE In T2 mode: The sync divider is set larger than 1. This is probably not intended. The sync divider is designed primarily for high sync rates from lasers and requires a fixed pulse rate at the sync input. In that case you should use T3 mode. If the signal at the sync input is from a photon detector (coincidence correlation etc.) a divider > 1 will lead to unexpected results. There are rare measurement scenarios where this condition is intentional and the warning can be disabled. In histogramming and T3 mode: If the pulse rate at the sync input is below 82 MHz then a divider >1 is not needed. The measurement may yield unnecessary jitter if the sync source is not very stable.	√	√	√
WARNING_DIVIDER_TOO_SMALL The pulse rate at the sync input (after the divider) is higher than 82 MHz. This is close to the sustainable front end speed. Sync events may be lost above 85 MHz. To avoid this, increase the sync divider.	√		√
WARNING_TIME_SPAN_TOO_SMALL This warning is issued in histogramming and T3 mode when the sync period (1/SyncRate) is longer than the start to stop time span that can be covered by the histogram or by the T3 mode records. You can calculate this time span as follows: Span = Resolution * Length Length is 32768 in T3 mode. In histogramming mode it depends on the chosen histogram length (default is 65536). Events outside this span will not be recorded. There are some measurement scenarios where this condition is intentional and the warning can be disabled.	√		√
WARNING_OFFSET_UNNECESSARY This warning is issued in histogramming and T3 mode when an offset >0 is set even though the sync period (1/SyncRate) can be covered by the measurement time span (see calculation above) without using an offset. The offset may lead to events getting discarded. There are some measurement scenarios where this condition is intentional and the warning can be disabled.	√		√

Warning	Histo Mode	T2 Mode	T3 Mode
WARNING_COUNTS_DROPPED This warning is issued when the front end of the data processing pipeline was not able to process all events that came in. This will occur typically only at very high count rates during intense bursts of events.	√	√	√
WARNING_USB20_SPEED_ONLY This warning appears when the HydraHarp 500 's USB connection is running only at USB 2.0 speed. For proper performance it should be running at USB 3.0 super speed. Check the cabling and the USB port in use. The same issue is indicated by the USB status LED showing yellow instead of green.	√	√	√

If any of the warnings you receive indicate wrong pulse rates, the cause may be inappropriate input settings, wrong pulse polarities, poor pulse shapes or bad connections. When in doubt, check all signals with an oscilloscope of sufficient bandwidth.

Note that the software can detect only a subset of all possible problematic measurement conditions. It is therefore not safe to assume “all is right” just by seeing no warning. On the other hand, if any of the warnings turns out to be an unnecessary nuisance, e.g., because your specific measurement conditions inevitably trigger it, you can disable that warning via the General Settings dialog (see section 7.7). Note, however, that disabling the warnings will also apply to which warnings you get to see in the Curve Details dialog when an existing data file is inspected. It may be better then to enable all warnings.

9.2. Data File Formats

While for many purposes the ASCII export of histograms to files or to the clipboard is sufficient and easy, you also may want to access the HydraHarp 500 data files via custom programs. This section provides only a brief overview on the file format. For details please refer to the online help file available via the help menu.

To overcome certain limitations of various different formats used in the past, PicoQuant now uses a unified file format. It is designed to be future poof in the sense that files created by a current software version stay valid for future software revisions and, moreover, files created by future software versions will most likely still be readable by older software, although they might contain information, that software can't even "know" about. This is achieved by using a tagged format. Tags identify the data to follow, and give the type, length and even meta information. The exact location of an individual item in the file is then irrelevant. Version robustness is granted as long as version-breaking changes to the semantics of a given field are implemented by a tag with a new identifier rather than expanding the range or interpretation of the old one. The list of tags (identifiers) and their interpretation rules can be kept in a tag dictionary. With this as a precondition, the software only has to show tolerance on missing non-mandatory (i.e. optional) content.

The new format definition unifies PicoQuant's existing file formats which individually evolved over many years. The resulting new TTTR file format with the extension `*.ptu` will be used for all current and future TCSPC products supporting TTTR mode and enriches them with powerful new features. Similarly, a tagged file format with the extension `*.phu` covers the histogram data formats of our current and future TCSPC products.

To support understanding of the format and implementation of custom software accessing these files, a set of demos is provided in the subfolder `\Filedemo` in your chosen software installation folder. If you need to evaluate more header items than the demos do, please refer to the HydraHarp 500 online help available via the help menu. A file format related HTML help file is also provided in the file demo folder. It contains a list of tag types and a tag dictionary that explains the individual items. Note that the dictionary contains more items than the HydraHarp 500 software actually uses. It is recommended to go by a specific file, have one of the demos read it and then look at the list of header items you get. You can also use the PicoQuant File Info shell extension that will be installed by the HydraHarp 500 software setup to inspect individual header items of a `*.ptu` or `*.phu` file. Just right-click on the file in Windows explorer and select *Properties*.

Despite the intended version tolerance of the tagged format, for consistency and safe version checking the HydraHarp 500 data files still carry a format version number, which is now called *content version* and currently has the string value “1.0”. In order to identify a HydraHarp 500 data file as a file created by and to be used by the native HydraHarp 500 software there is a tag *assured content* which begins with the string “HydraHarp 500“. There is also a pair of tags creator name and creator version that identify the creating software. Programmers of custom software writing such files MUST USE THEIR OWN CREATOR NAME.

Note that despite our best efforts towards compatibility and version tolerance, file formats in future software releases are subject to change without notice.

9.2.1. Interactive Mode File Format

The standard HydraHarp 500 histogram data files created by the HydraHarp 500 software in interactive histogramming mode (`*.phu`) are tagged binary files which contain both the setup parameters and the actual histogram data. The latter can be present multiple times, i.e. multiple measurements can be stored in one file. Relevant settings are stored for each measurement separately. In order to identify a HydraHarp 500 data file as a file created by and to be used by the native HydraHarp 500 software, a program reading in these files can read the tag *assured content* which begins with the string “HydraHarp 500“. However, a piece of software aiming solely at retrieving the histogram data content can (and should) be tolerant about this tag and go for the pure histogram data. This tolerance will ensure compatibility for the future. Indeed, the demos in the subfolder `\Filedemo` in your chosen installation directory are following this tolerant approach. For more information on individual file tags and their content, please consult the online help file available via the help menu.

9.2.2. TTTR Mode File Format

HydraHarp 500 data files from T2 and T3 Mode (`*.ptu`) created by the HydraHarp 500 software are tagged binary files which contain both the setup parameters and the actual event data. There can be only one measurement per file. The setup data in the file header is similar to that in standard interactive mode files. In order to identify a HydraHarp 500 data file as a file created by and to be used by the native HydraHarp 500 software, a program reading in these files can read the tag *assured content* which begins with the string “HydraHarp 500“.

However, a piece of software aiming solely at retrieving the event record data content can (and should be) be tolerant about this tag and go for the pure event record data. This tolerance will ensure compatibility for the future. Indeed, the demos in the subfolder `\Filedemo` in your chosen installation directory are following this tolerant approach. For more information on individual file tags and their content, please consult the online help file available via the help menu. It is worth noting that the actual TTTR record data following the file header corresponds directly to the raw data obtained with custom programs using the HydraHarp 500 programming library.

9.3. Hardware Technical Data

9.3.1. Specifications

All information given here is reliable to our best knowledge. However, no responsibility is assumed for possible inaccuracies or omissions. Specifications and external appearance are subject to change without notice.

Input Channels and Sync

Number of Input Channels	HydraHarp 500 S 4 up to 8	HydraHarp 500 M 8 up to 16
Trigger principle	edge trigger (ETR) or constant fraction discriminator (CFD)	edge trigger (ETR)
ETR trigger edge		rising or falling
ETR input voltage operating range (pulse peak) optimum:		−1200 … 1200 mV abs. Amplitude 150 … 300 mV
ETR input voltage max. range (damage level)		± 2500 mV
ETR trigger level adjust		−1200 … 1200 mV
CFD trigger edge		falling
CFD input voltage operating range (pulse peak) optimum:		−1200 … -10 mV -300 … -150 mV
CFD input voltage max. range (damage level)		± 2500 mV
CFD discriminator level adjust		−1200 … 0 mV
Input impedance		50 Ω
Input trigger hysteresis		ca. 3 mV (default) or 35 mV (settable)
Required input pulse width		≥ 0.4 ns
Required input pulse rise ⁄ fall time		≤ 20 ns
Marker and control inputs		standard LVTTL, > 2 kΩ DC load

Time to Digital Converters

Base resolution (min. time bin width)	1 ps
Single measurement timing uncertainty (rms)*	typ. 2.5 ps
Start-stop timing uncertainty (rms)*	typ. 3.5 ps $+ 10^{-10} \cdot \Delta t$
Differential non–linearity	< 5% p.p. < 1% rms

* In order to determine the timing uncertainty or "jitter" it is necessary to repeatedly measure a time difference and to calculate the standard deviation (rms error) of these measurements. This is done by splitting an electrical signal from a pulse generator and feeding the two signals each to a separate input channel. The differences of the measured pulse arrival times are then calculated along with the corresponding standard deviation. This latter value is the rms jitter or timing uncertainty of start-stop measurements. However, calculating such time differences requires two measurements. These both have a small random jitter and a term that is dependent on the time difference between the two signals because over longer spans the crystal clock jitter becomes noticeable. For short term measurements on a sub-microsecond scale, as in most TCSPC applications, the latter is typically negligible. The timing precision of a single measurement, according to error propagation laws, is obtained by dividing the start-stop timing uncertainty by √2. We specify this single measurement rms error here for comparison with other products. Note: These specifications apply for the current hardware and gateware of August 2025.

Dead time	< 680 ps (ETR)* < 6.8 ns (CFD)
Peak count rate	$1.25 \cdot 10^9$ cps sustainable for bursts of 2048 events
Max. sustained count rate	$85 \cdot 10^6$ cps
Max. sync rate without divider	85 MHz
Max. sync rate with divider (periodic sync)	1200 MHz (ETR) 140 MHz (CFD)
Adjustable delay range for each input	± 100 ns

* events following in rapid succession may also be lost due to speed limits of the front end FIFOs, see specification of peak count rate

Histogrammer

Maximum number of time bins		131072 *
Full scale time range (depending on chosen resolution)		131 ns … 1.1 s
Count depth per time bin		4,294,967,296 (32 bit)
Acquisition time		hardware timer 1 ms … 100 h or infinite under software control
Sustained throughput to device memory (total for max. number of possible channels)	HydraHarp 500 S $180 \cdot 10^6$ cps	HydraHarp 500 M $360 \cdot 10^6$ cps **

* limited to 65536 in the standard HydraHarp 500 software
** $180 \cdot 10^6$ cps in each of two groups of input channels, where group 1 = ch {1,2,5,6,9,10,13,14} and group 2 = ch {3,4,7,8,11,12,15,16}

TTTR Engine

T2 mode resolution	1 ps
T3 mode resolution (settable)	1 ps … 8.38608 ms
T3 Mode number of time bins	32768
FIFO buffer depth (records)	134,217,728
Sustained throughput over USB (sum of all channels)	typ. 85×10^6 events/sec (depending on host PC configuration and performance)
Acquisition time	hardware timer 1 ms … 100 h or infinite under software control
External marker inputs (LVTTL)	4
External marker input pulse duration	> 50 ns
External marker input pulse spacing	> 50 ns
External marker rise/fall time	< 50 ns
External Marker timing error in T2 mode	< 100 ns
External Marker timing error in T3 mode	< Tsync + 100 ns

Clock Sources and Output

Internal clock	oven controlled crystal oscillator (OCXO) frequency accuracy ± 300 ppb frequency stability (ambient temperature 0..35°C) ± 10 ppb
Reference clock input	10 MHz 200 … 1 500 mV p.p. 50 Ω; AC coupled

Reference clock output	Default: 10 MHz White Rabbit mode: 31.25 MHz 250 mV p.p. 50 Ω; AC coupled

Trigger Output

Baseline level	0V typ.
Pulse amplitude	-0.7V typ.
Pulse duration	10 ns typ.
Load/Termination	50 Ω
Programmable period range	0.1 µs ... 1.678 s

Operating Environment

Environment restrictions	Indoor use only
Ambient temperature	0 .. 35°C
Relative humidity	< 80% non-condensing
Operation altitude	max. 2000 m above sea level
Warm–up period to meet specs	20 min
Recommended PC specifications	
CPU	≥ 2 cores, ≥ 2 GHz
RAM	≥ 8 GB
USB minimum required	USB 2.0
USB required to meet specs	USB 3.0 or compatible higher
Operating system	Windows 10 or 11 or Linux with Wine (see section 9.4)

Power Supply

Line voltage	100...240V AC 50...60 Hz
Power consumption	≤ 50 W
Fuse	2 x T1.6A/250V

Dimensions

Housing incl. feet and handles	305 × 350 × 95 mm

Retraction of Discarded Devices

Waste electrical products must not be disposed of with household waste.

This equipment should be taken to your local recycling center for safe treatment.

WEEE–Reg.–Nr. DE 96457402

9.3.2. Connectors

The inputs for the photon detector signals and the sync signal are SMA connectors located on the front panel of the HydraHarp 500. They are labeled **SYNC, CH1..CHx** (x depending on the model). The inputs are terminated with 50 Ohms internally. Use quality 50 Ohms coax cables with appropriate connectors. For interfacing to BNC connectors use standard adapters. Carefully screw on the SMA connectors for sync and detector until they are hand-tight. Do not use wrenches. Note that PMT detectors operate with high voltages that may discharge through the signal cable. Make sure such detectors are switched off and fully discharged before connecting them.

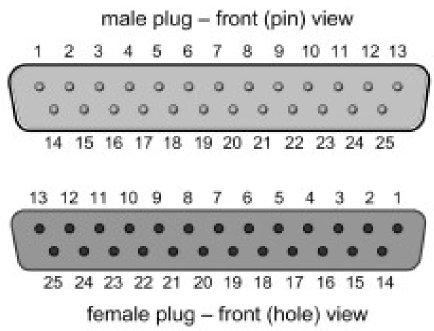
Apart from the SMA connectors for the input signals the HydraHarp 500 has further connectors for USB, White Rabbit, clock synchronization, and experiment control.

The **USB connector** (blue) is a standard USB 3.0 type B socket located at the front panel. It is used to connect to the host PC via a qualified USB 3.0 cable. Observe the notes in section 4.5 for making this connection.

The **White Rabbit (WR) connector**, also on the front panel, is designed to hold exchangeable SFP transceiver modules. Upon shipping such a module is not installed. Suitable SFP transceiver modules can be ordered from PicoQuant as well as from other vendors. The SFP port can optionally also be used to connect an external FPGA and stream data to/from it. White Rabbit is in this case disabled. For details on the External FPGA Interface (EFI) please see the separate manual. More documentation and demo code is provided as part of the EFI gateware and software pack. The most recent EFI pack can be downloaded from the PicoQuant downloads page at https://www.picoquant.com/downloads.

The **control connector** is a 25-pin female sub-D connector labeled 'CTRL' and located at the back of the housing. Note that future firmware/software is going to allow reconfiguration of some of these pin assignments.

The following figure shows the pin layout and the table below contains the connector's default pin assignments.



CTRL Connector – Pin numbering scheme

CTRL Connector – Default Pin Assignments			
Pin#	**Name**	**I/O**	**Purpose/Description**
1	GPIO 0	LVTTL in	marker 1 input
2	GPIO 1	LVTTL in	marker 2 input
3	GPIO 2	LVTTL in	marker 3 input
4	GPIO 3	LVTTL in	marker 4 input
5	GPIO 4		reserved
6	GPIO 5		reserved
7	GPIO 6		reserved
8	GPIO 7		reserved

9	GPIO 8		reserved
10	GPIO 9		reserved
11	GPIO 10		reserved
12	GPIO 11		reserved
13	GPIO 12		reserved
14	GND	GND	Ground (0V)
15	GPIO 13		reserved
16	GPIO 14		reserved
17	GPIO 15	LVTTL in	RS232 RX (3.3V signal level) for connection to GPS/GLONASS receiver
18	GND	GND	ground (0V)
19	C1	LVTTL in	start measurement (requires dedicated software)
20	C2	LVTTL in	stop measurement (requires dedicated software)
21	GND	GND	ground (0V)
22	MACT	LVTTL out	high when measurement running
23	GND	GND	ground (0V)
24	D3V3	DC out	+3.3V / ≤ 350 mA supply for external hardware add-ons
25	GND	GND	ground (0V)

The specifications for LVTTL signals can be found at http://wikipedia.org/wiki/Transistor-transistor_logic. Note that for the hardware described here a maximum high level of 3.3V is permitted.

For further details regarding signal specifications, notably the trigger output (voltages, frequencies pulse widths etc.) see section 9.3.1.

Pins 1, 2, 3 and 4 accept LVTTL compatible synchronization signals that will be recorded as markers in TTTR mode. The pins are internally pulled down, so that they are inactive when left unconnected. The active edge is chosen in the software settings dialog. Rise/fall times must be 50 ns or faster. Both high and low state must be at least 50 ns long. The clock period may therefore (in principle) be as short as about 100 ns but data bus throughput constraints will apply. Each marker creates an additional TTTR record, so that one must ensure not to swamp the data stream with too many marker records. When bandwidth gets tight, markers take precedence over photon records, so that excess marker traffic can suppress photon records. In fast imaging applications it is therefore recommended not to use a pixel clock but a line clock only. Because each photon has a time tag, it is usually not necessary to use an additional pixel clock. For more information on how to use the marker inputs see section 6.3.5.

Pin 17 can be used to connect the serial TX line of a GPS/GLONASS receiver for time transmission via the NMEA ZDA Message protocol at 115.2 kbps, 3.3V. We have tested this with the Mini-T™ GG disciplined clock module from Trimble Inc., USA. See section 7.7 for suitable clock source selection.

Pins 19 and 20 can be used to implement hardware triggered measurements. Note that this requires custom software (see the DLL manual and related demos).

Pin 22 is a LVTTL output that goes high when a measurement is running.

Pin 24 provides a 3.3 V DC supply voltage that external electronics can use. Under no circumstances must this line be shorted to ground or loaded in excess of the specified maximum current.

Pins 14, 18, 21, 23, and 25 are the common ground for the LVTTL signals and the DC supply line. They are also connected to the housing.

Make sure not to confuse the control connector with a connector for other equipment that would physically fit but might lead to mutual damage. Appropriate cables for typical applications of the control port are available from PicoQuant.

At the back of the housing of the HydraHarp 500 there is a standard **connector for power supply** that includes two fuses. The fuses are the device's only user serviceable parts. Please find the required fuse current ratings on the product label or in section 9.3.1. Do NOT replace the fuses with types of higher current rating than specified. Use slow blowing fuses of the specified rating only. If the fuse blows repeatedly stop further experiments and contact support.

At the back there also another four SMA connectors labeled “TRG OUT”, “REF IN”, “REF OUT”, “PPS IN”.

TRG OUT is a programmable trigger output for pulsed light sources. Note that this is under software control and must be used with greatest care when potentially dangerous laser sources are triggered from it. The trigger rate is programmable from 0.1 µs to 1.678 s (0.596 Hz to 10 MHz). The pulse width is about 10 ns, the base line level is near 0 V and the active level (pulse peak) is approximately -0.7 V. When a light source is triggered from this signal the same signal must typically be fed to the sync input. This requires proper impedance matching e.g., by means of a power splitter (reflection-free T-pad).

REF IN accepts an industry standard 10 MHz clock reference, e.g., from a frequency normal, an atomic clock, or another HydraHarp 500. When this clock source is selected (via software) then the HydraHarp 500 will lock its internal PLL to this signal. Note that it must not be disconnected once it is being used as the active clock source.

REF OUT by default provides an industry standard 10 MHz clock output phase locked to the HydraHarp 500 's internal clock, e.g., for the REF IN of another HydraHarp 500. Once the receiving device has locked its PLL to this signal, the clocks of the two devices will not drift apart. If the HydraHarp 500 is operated in White Rabbit mode the pin will output a different clock frequency, namely the White Rabbit system clock of 125 MHz divided by 4, i.e. 31.25 MHz.

PPS IN accepts the “pulse per second” signal of e.g., GPS receivers. In combination with the 10 MHz clock and the time code transmitted by the GPS receiver it is possible to synchronize the HydraHarp 500 's clock and time from GPS. This can be used to synchronize two or more HydraHarp 500. The accuracy of such a synchronization depends on that of GPS and the quality of the receiver. It is typically on the order of some tens of ns. The benefit is remote synchronization over large distances. If better accuracy is required, consider White Rabbit instead.

9.3.3. Indicators

9.3.3.1. Device Status LEDs

The HydraHarp 500 shows some status information by means of three LEDs on the front panel labeled STATUS. The meaning of these indicators, from left to right, is as follows:

USB Status (left)

red = no connection
orange = USB 2.0 connection
green = USB 3.0 connection

Measurement Status (middle)

red = measurement active
green = inactive

This reflects the state of pin 22 of the CTRL connector (high when measurement running).

Error Status (right)

red = error (or not yet initialized)
orange = info
green = OK

Note that the error status LED is initially red when the device is switched on and the software has not yet initialized it. Also note that it gives only a very crude indication of errors. The indicated errors or info states are also monitored by the software which gives much more tangible information on the issue. The most likely info you may see indicated by orange light is when the input rate is too high and events are dropped. However, especially in case of infos the LED may actually light up only very briefly. Looking at the status information given by the software should therefore always be preferred.

9.3.3.2. Input Channel Status LEDs

The HydraHarp 500 shows the status of the input channels by means of multi-color LEDs above the SMA input connectors. The meaning of the colors is as follows:

LED Color	**Interpretation**	**Cause and Comments**
off	channel not enabled	Indicates that the device was purchased as an economy version with only a subset of channels enabled. An upgrade to enable such channels can be ordered at any time later.
dimmed purple	channel disabled by software	Check software settings if this was not intentional.
green	channel is in ETR mode and ready but no signal is detected	ETR = Edge Trigger No signal and inappropriate trigger level has the same effect here.
blue	channel is in CFD mode and ready but no signal is detected	CFD = Constant Fraction DIscriminator No signal and inappropriate discriminator level has the same effect here.
white	channel ready and signal de-tected	May flicker when this state is transient.
orange	channel ready, signal detected but events are being dropped	The input rate is so high that the input circuits cannot keep pace and events are dropped. May flicker when this state is transient.

9.4. Using the Software under Linux

The HydraHarp 500 software can also be used under Linux (x86-64 platform only). This requires that Wine is installed (see https://www.winehq.org). You can run the regular software setup as explained in section 4.4. Instead of installing a device driver, running under Linux with Wine requires that you have Libusb 1.0 installed (see https://libusb.info/). We have successfully tested the following configurations:

- Wine 8.0 and Libusb 1.0.25 on Ubuntu 22.04 LTS.
- Wine 9.16 and Libusb 1.0.27 on Ubuntu 24.04 LTS
- Wine 9.0 and Libusb 1.0.27 on Linux Mint 22

In all cases we used the package `winehq-stable` which is typically more mature than the version provided by the Linux distribution.

Access Permissions

For device access through libusb suitable permissions for the device must be granted to the normal user, otherwise only the super-user `root` will have access. Recent Linux distributions use udev to handle this. For automated setting of the device access permissions with udev you can add an entry to the set of rules files that are contained in `/etc/udev/rules.d`. Udev processes these files in alphabetical order. The default rule files usually carry names starting with a number. Don't change these files as they could be overwritten when you upgrade your system. Instead, put your custom rule for the HydraHarp 500 in a separate file. The typical content of this file should be:

```
ATTR{idVendor}=="0d0e", ATTR{idProduct}=="0016", MODE="666"
```

A suitable rules file `HydraHarp500.rules` is provided in the folder `udev` under the distribution pack folder `HydraHarp500 v.X.X.X.X\installation\Linux`. The install script in the same folder does just this. Note that this requires root permissions. As a normal user you must run it preceded with `sudo`. After that you need to disconnect and reconnect the device to get access.

If you have issues obtaining permissions recall that the name of the rules file is important. Each time a device is detected by the udev system, the files are read in alphabetical order until a match is found. Different Linux distributions may use different rule file names for various categories. If there happen to be later rules that are more general (applying to a whole class of devices) they may override your custom rule and the desired access rights. It is therefore important that you use a rules file named such that it gets evaluated after the general case. The default naming `HydraHarp500.rules` most likely ensures this but if you see problems you may want to check.

Note that the setting `MODE="666"` is quite permissive for all users. If you prefer tighter security regarding device access please study the documentation of udev and/or the recommendations of your distribution for handling USB device access, e.g. employing user classes with suitable access rights.

Wine Limitations

Please note that running the software under Wine is an experimental feature with limited support. You may observe failure or some glitches that come and go when using the next version of Wine. The latest may or may not be the most stable.

9.5. Citation

If you have obtained exciting results with one of our systems, please let us know, and where appropriate, please mention the instrument in your publications.

At our web–site we also maintain a large bibliography of publications related to our instruments. It may serve as a reference for you and other potential users. See http://www.picoquant.com/scientific/references. Please kindly submit your publications for addition to this list.

PicoQuant GmbH
Rudower Chaussee 29 (IGZ)
12489 Berlin
Germany

P +49-(0)30-1208820-0
F +49-(0)30-1208820-90
info@picoquant.com
http://www.picoquant.com