IKA LOGIC
Augmented Instruments

Logic analyzers user's manual

ScanaStudio V4.0 and later

Ibrahim KAMAL

December 1, 2019

Contents

Logic Analyzers manual

What is a logic analyzer

There is a lot of history around logic analyzers. While Wikipedia covers the early days of logic analyzers in this page https://en.wikipedia.org/wiki/Logic_analyzer, we're going to present how the logic analyzers have evolved to what they are today.

Logic analyzers are measurement devices - just like oscilloscopes or voltmeters - that are used to measure the variation of **logic level** over a period of time. By logic level, we imply that we're only interested in knowing the binary state of a digital signal: 0 or 1, and unlike oscilloscopes, logic analyzers are not designed to look at analog voltages (like a sine wave).

Now to be fair, all electronic signals are "analog signals" and there is no such thing as a pure logic signal without any noise or overshoot. Although a logic analyzer will display captured signals as perfect square shaped signals, complicated circuits in the logic analyzer hardware convert the analog signals to digital (logic) signals.

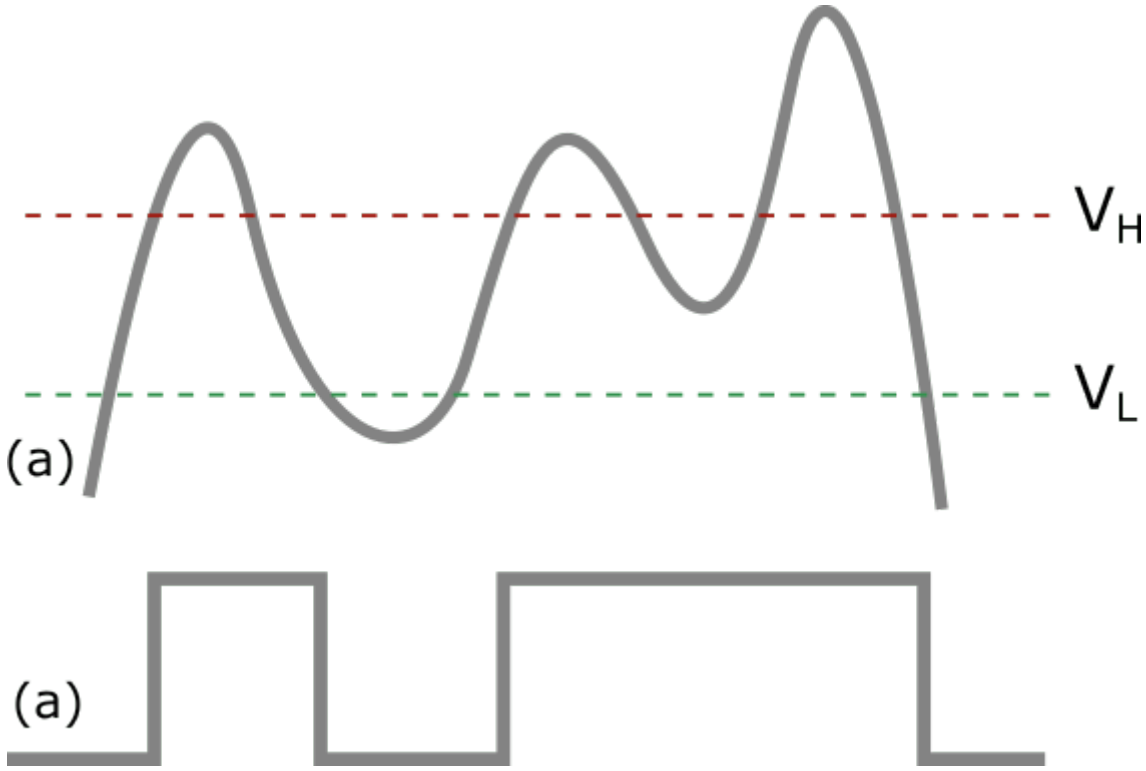


Figure 1: analog vs analog signals

The image above shows how an analog signal (a) is converted to a digital signal (b). typically, the frond end circuit of a logic analyzer will implement some kind of hysteresis using two thresholds (V High and

V Low) to filter out glitches and noise.

While a logic analyzer will typically store much less details about a signal compared to an oscilloscope, it will capture and store signals for a much longer period. That's what we call "capture depth". Some logic analyzer will even capture signals for an unlimited time, that is, until the user manually stops the capture.

Logic analyzer applications

A logic analyzer is used to diagnose and debug digital circuits and embedded systems. Logic analyzers have became *must have* tools if you're working with micro-controllers, FPGAs or other electronic circuits, as it will allow you to visualize the signals generated by a firmware, by a sensor or any similar digital circuit. Typically, one will use a logic analyzer to ensure that components are behaving correctly - e.g. an I2C device is responding to a micro controller requests - and rectify the embedded firmware in consequence.

Below is an example capture of RS232 (Serial UART), captured by an Ikalogic logic analyzer device, and decoded using ScanaStudio software:

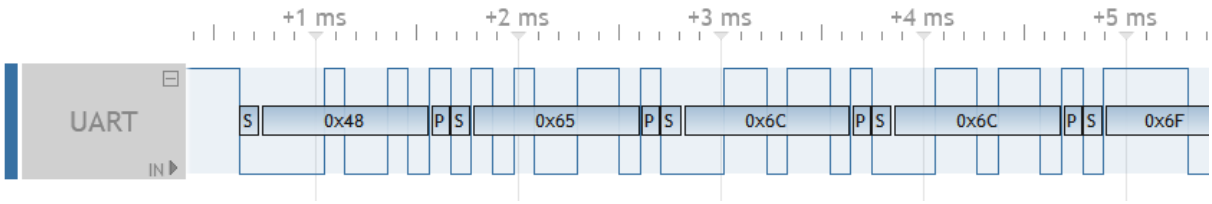


Figure 2: UART (RS232) capture and decode logic analyzer

Obviously, logic analyzers can also be used for reverse engineering applications.

Capture depth is one of the major features that characterizes a logic analyzer as it helps understanding the full chain of events that led to a particular behavior or event.

Ikalogic devices

Ikalogic provides logic analyzers that connect to a computer via a USB cable. A free software - called ScanaStudio - is used to configure the device and display captured signals. The software can also be used to further analyze the captured samples by decoding protocols like I2C, SPI or UART.

ScanaStudio software

ScanaStudio is a free software used to view and analyze signals captured with Logic Analyzer devices. Devices are connected to host computer via USB and transfer captured data to ScanaStudio. Some devices stream data in real time (like SP209 series logic analyzer), other devices (like SQ Series store data in its internal memory buffer before transferring data after capture is done.

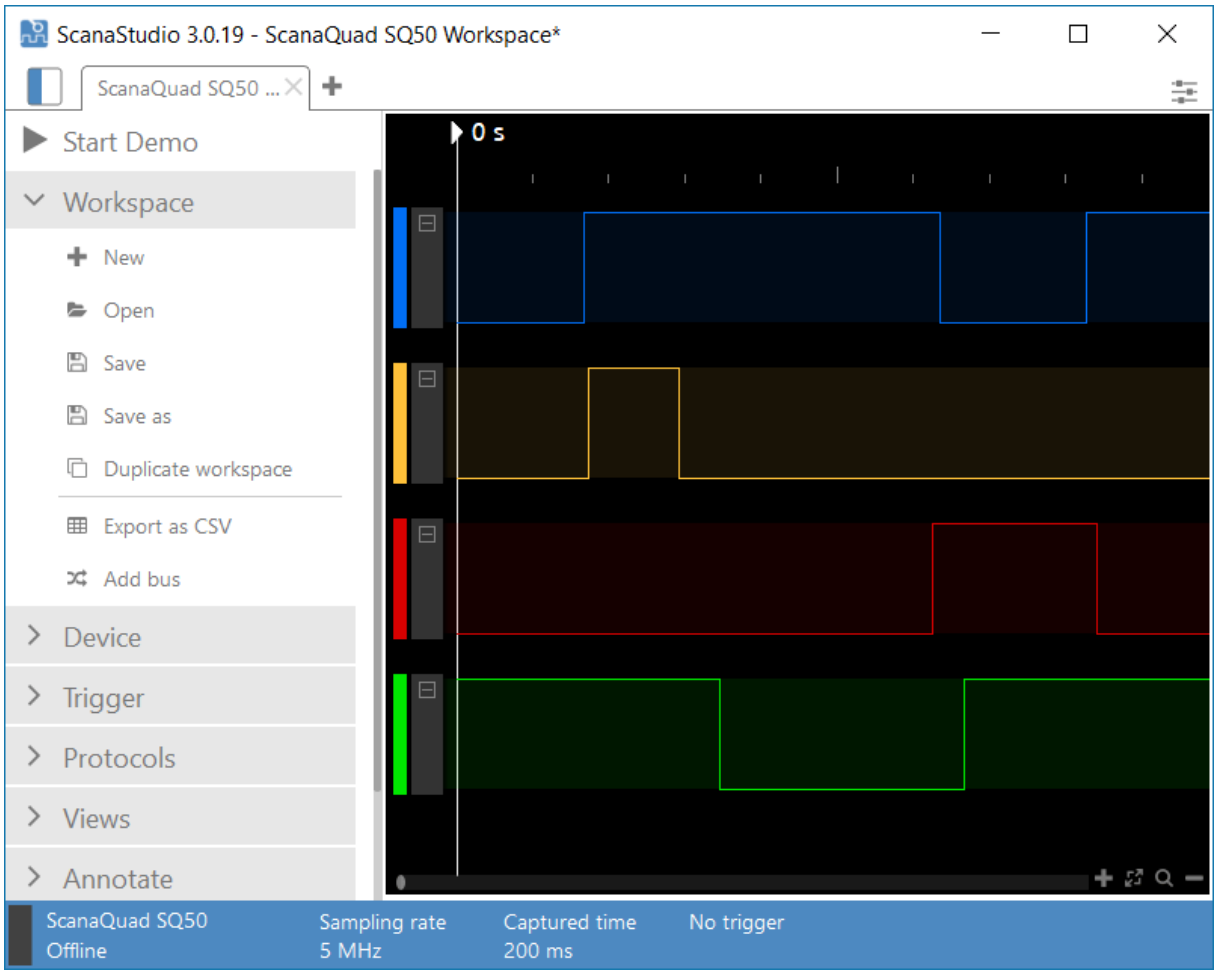


Figure 3: ScanaStudio logic analyzer software

ScanaStudio is also used to configure the logic analyzer device, like setting up the trigger conditions and configuring parameters like the sampling rate and the number of samples.

For some devices that have the ability to generate samples (like SQ series), ScanaStudio is also used to build the logic patterns to be generated.

In other terms, ScanaStudio is *the* user interface of all Ikalogic Logic Analyzers. For this reason, this manual will mainly address the software part.

Along the years, ScanaStudio have evolved to offer highly advanced features like protocol decoding, signals editing, packetized view and data filtering.

Installing ScanaStudio

The latest version of ScanaStudio can be downloaded here: https://www.ikalogic.com/scanastudio-logic-analyzer-software/

Windows

The downloaded windows package consists of a setup program that needs to be launched. If asked to grant administrative right to that application, please accept. To uninstall, simply use window's "Add/Remove programs" wizard.

Linux

The linux package needs to be uncompressed to a folder of your choice. Then, inside this folder you'll find an installer script that needs to be launched. This script will usually ask for SUDO password to proceed with installation. The script will execute a series of commands, then notify you when it's done. If any error message appears, please send a mail to our support services with the whole output of the script. Once installed, on Ubuntu (and similar) distributions, you can open ScanaStudio from the Unity Launcher (quicklaunch).

MacOS (OX X)

ScanaStudio for macOS comes bundled in a standard DMG archive. To install it, double click the .dmg file to make its content available (name will show up in the finder sidebar). Usually a window opens, showing the content as well. Drag the application from the .dmg window into /Applications to install (may need an administrator password).

Introduction to SP209 series logic analyzers

What are SP209 series?

SP209 series logic analyzers and protocol decoders offer in depth analysis of logic signals and protocols with 200MHz (5ns) timing resolution. 9-channel operation allows 8-bit parallel data to be captured along with a clock or strobe signal. SP209 range of logic analyzers is composed of two devices, SP209 and SP209i. Both offer 9 logic inputs, but SP209i integrates industrial receivers for CAN, LIN, RS232 and RS485 buses.

Below is a top view of the SP209i Logic Analyzer:

Figure 4: SP209 Series logic analyzer (industrial version)

Principle of operation

SP209 Series logic analyzers connects to a computer via a USB cable. A free software - called ScanaStudio - is used to configure the device and display captured signals. The software can also be used to further analyze the captured samples by decoding protocols like I2C, SPI or UART.

Embedded memory vs streaming USB based logic analyzers (ones that don???t have a display and rely on a computer for that matter) usually operate according to one of two schemes:

- Using an embedded memory to store captured samples. Samples are later downloaded at a slower speed via the USB interface. This has the advantage of not being limited by USB transfer rate, but has the disadvantage of limited embedded memory.
- Streaming captured samples over the USB connection, at the maximum possible speed. While this offers the advantage of a virtually unlimited memory (only limited by host computer???s

memory), it has the disadvantage of limiting the sampling rate to USB???s throughput.

SP209 combines the advantages of both streaming and embedded memory techniques. An embedded 2Gb DDR-3 memory stores captured samples at 200 MHz sampling rate on all channels, while a USB interface compresses and transfers the data simultaneously, effectively emptying the embedded memory and making more room for new samples. This results is a logic analyzer that can capture dozens of minutes of logic signals activity on 9 channels at 200 MSPS.

The diagram below shows the flow of sampled signals:

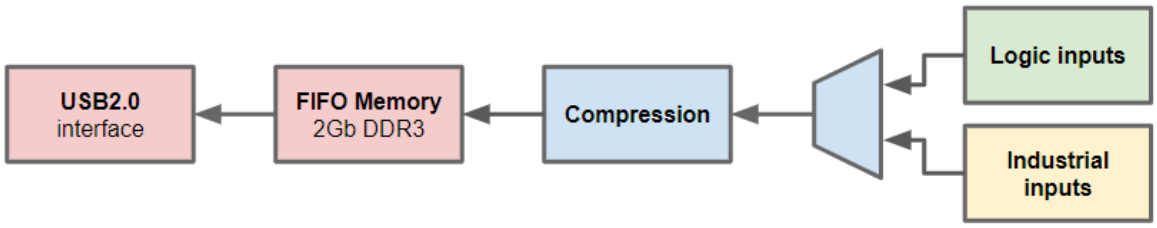


Figure 5: SP209 logic signal compression and streaming

Typical applications

SP209 series logic analyzers are perfectly adapted for demanding applications, where it is needed to capture logic signals with maximum time resolution on all 9 channels.

- Embedded systems debugging
- Research and education
- Serial protocols analysis, like I2C, SPI, UART or 1-Wire (non exhaustive list)
- Industrial bus analysis: RS232, RS485, CAN, LIN
- ADC diagnostic

On top of standard logic analysis, the industrial version (SP209i) can be used to analyzer industrial buses. Furthermore, one can use both the industrial inputs and logic inputs to capture the signals before and after an onboard transceiver. This can help characterizing a system like propagation delays or transceiver intermittent failures.

Package content

SP209 series logic analyzers ship with the following accessories:

- USB cable (micro-B to A)

Figure 6: USB cable

- 10 Logic probes (9 signals + 1 ground)

Figure 7: logic-probes

- 10 slots cable comb (to group logic probes together)

Figure 8: Cable-comb

- SMA anti-dust covers

Figure 9: SMA-cover

Device ports

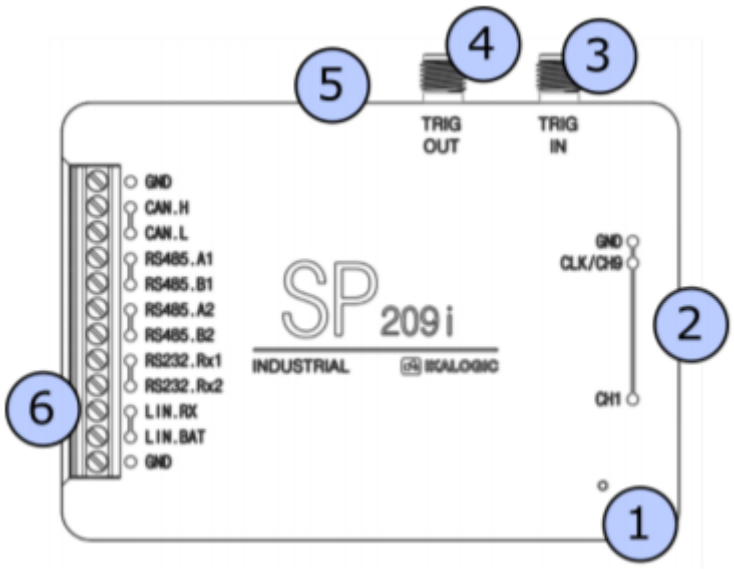


Figure 10: sp209 ports

SP209(i) logic analyzer have the following ports and interfaces:

1. Status LED
2. 9-CH logic probes input
3. Trigger OUT SMA connector
4. Trigger IN SMA connector
5. USB (micro-B) port.
6. Industrial port (only on industrial version)

Device datasheet

For detailed timing and DC characteristics, power consumptions and input characteristics, please download latest version of the datasheet

SP209 series device drivers

Windows OS SP209 drivers are included and should be automatically installed with ScanaStudio software. Because certain system configurations may require manual installation, the following sections provide guide lines to perform driver install or update for SP209.

Manual driver setup on windows To setup the drivers manually, follow the following steps:

1. Download the driver package from this url: https://cdn.ikalogic.com/dist/drivers/ScanaStudioDriver.zip
2. Unzip the driver package to a known folder, (e.g. `c:\ScanaStudioDriver\`)
3. Connect the SP209 to a USB port on your computer (**Important**: avoid using any HUBs, connect directly to root USB port.)
4. Open device manager and locate the SP209 device in the list. The device may be named "IKALOGIC Device", "IKALOGIC COM PORT" or "USB Serial port" (even though SP209 is not a serial port, until the driver is correctly installed, windows wont be able to correctly enumerate it). You may also disconnect and reconnect the SP209 device to see which device appears and disappears in the list. It's also worth noting that if the SP209 driver is not correctly installed, a yellow exclamation mark should be visible next to the device name in the list
5. Once the device is identified, right click on it and select "Update driver" (or an equivalent item depending on the language of your system).
6. Follow the steps in order to specify the location of the driver on your computer.
7. Provide the path to the drivers folder that was created in step 2.
8. Follow the lasts steps in the wizard to finish installing the driver.

Linux and MacOS Linux and MacOS do not need any specific operations to install SP209 device on the system. The most important part is to ensure the SP209 is directly connected to a root USB hub, and not via a HUB.

In case of doubt, or if any diagnostic is needed, the SP209 device uses the following USB VID / PID pair:

- VID: 0x0403
- PID: 0x7FD0

SP209 uses an FTDI chip, loaded with a proprietary PID that is specific to Ikalogic products. This prevents FTDI drivers from claiming the device as an FTDI serial COM port.

SP209 LED indications

SO209 status led can either be Off, White, Orange or blinking.

Led state	Meaning
Off	Device not powered (not connected to a USB port)
Orange	Device connected to USB port but not communicating with ScanaStudio software

Led state	Meaning
White	Device connected to USB port and communicating with ScanaStudio software
White blinking	Activity (logic state change) is detected on one or more of the logic channels

Introduction to SQ series logic analyzers

What are SQ Series?

SQ Series logic analyzers are entry level devices capable of capturing and generating logic signals at maximum sampling rate of 200 MHz on 4 channels. SQ range of logic analyzers is composed of 4 devices:

- SQ25
- SQ50
- SQ100
- SQ200

SQ series of logic analyzers are small devices (50mm x 50mm x 25mm) as seen in the image below:

Please refer to product page for detailed features and limitations of each device.

Principle of operation

SQ Series logic analyzers and pattern generators connects to a computer via a USB cable (mini-B). A free software - called ScanaStudio - is used to configure the device, display captured signals and build signals to be generated. The software can also be used to further analyze the captured samples by decoding protocols like I2C, SPI or UART. It's worth noting that SQ series can capture *and* generate signals simultaneously, and ScanaStudio software is perfectly adapted for that type of applications.

Embedded memory architecture Unlike other devices that stream data over USB, SQ series uses an internal memory to store captured samples (or samples that need to be generated).

To get the most out of your SQ device, it's important to have a global understanding of how its internal memory works.

SQ's memory can be divided into two sectors:

- Capture memory
- Generator memory

The amount of memory allocated to capture or generation can be defined without much restrictions. The animation below shows how the memory organization affects device operation:

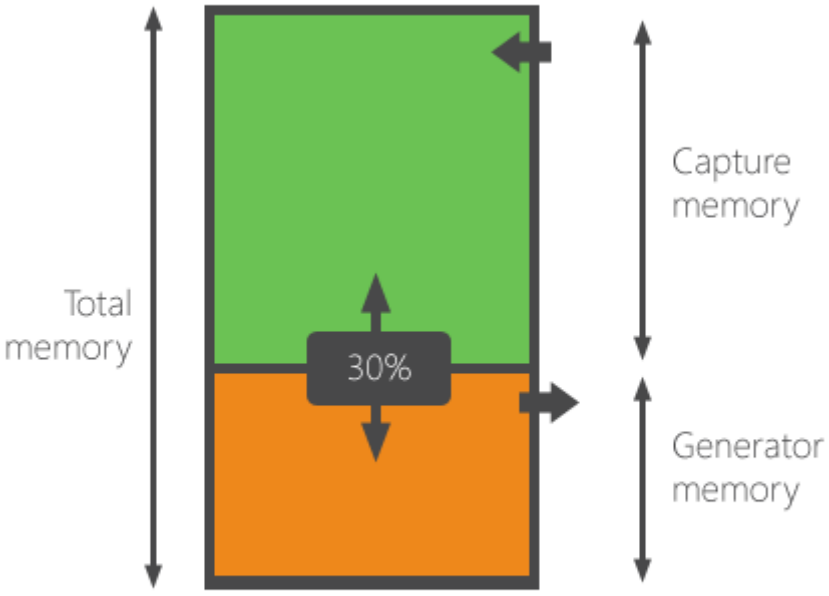


Figure 11: sq logic analyzer memory organization

While capture memory affects the total capture time, generator memory defines the total length of the signal that can be generated. In case you want to generate a signal in a loop, the total generator memory also defines the total period (and hence frequency) of your generated signal.

Capture mode If your ScanaQuad device is configured in capture mode, the total memory is available for the captured samples. However, you may only allocate a part of that memory for signal capturing.

Generator mode If your Scanaquad device is configured in generator mode, the total memory is available for the generator. Samples will be transferred to ScanaQuad memory and then generation will start.

The sample sequence may be generated in a loop or only once depending on the check box “Loop generated signal”:

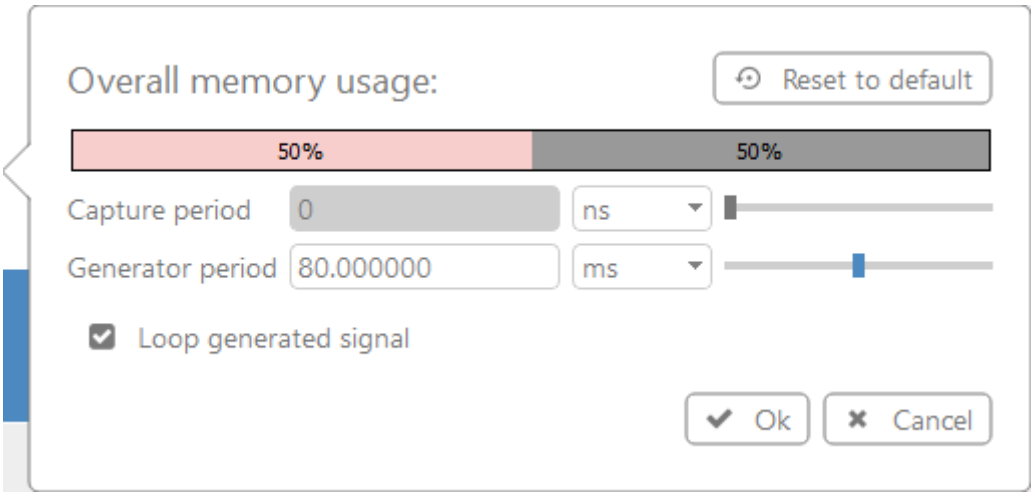


Figure 12: sq memory configuration

Note: Both capture and generator mode share the same internal sampling clock, so it is not possible to have a different clock for capturing and generating signals.

Typical applications

SQ series logic analyzers are meant to be versatile devices as they are capable of capturing and generating logic signals. The small number of channels (4 channels) make it ideal for low-budget applications where serial protocol diagnostic is needed. Among most typical applications, we can list the following:

- Embedded systems debugging
- Testing
- Research and education
- Serial protocols analysis, like I2C, SPI, UART or 1-Wire (non exhaustive list)

Being able to capture and generate signals simultaneously makes SQ series ideal for application where it is needed to simulate a circuit a capture its response.

Package content

SQ Series logic analyzer is shipped with the following accessories:

- 5 points hook-probes set:

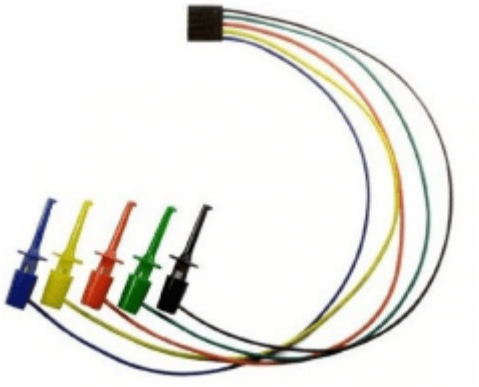


Figure 13: sq probes set

- 0.8 meter USB cable (mini-B to A):

Figure 14: SQ USB cable

Device ports

SQ logic analyzer and pattern generator have the following ports and interfaces:

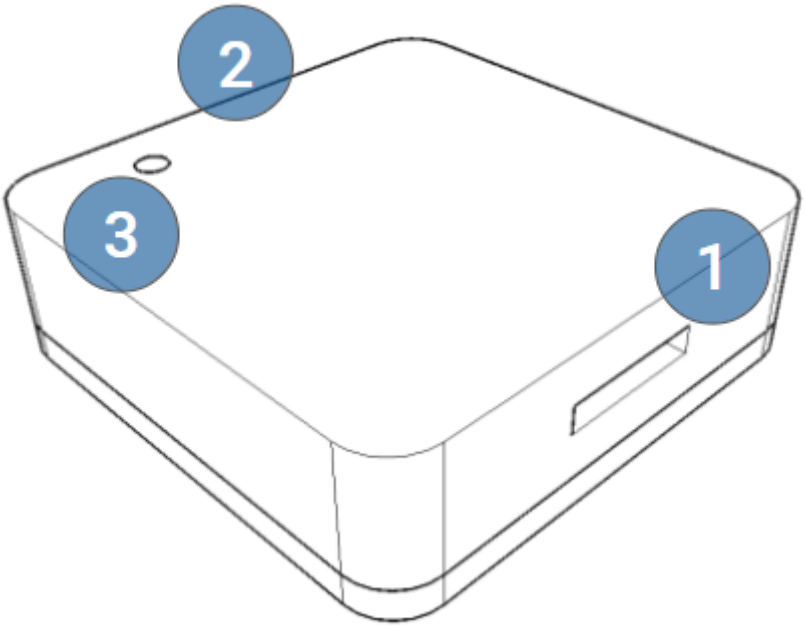


Figure 15: sq device ports

1. 4-CH logic probes input (4 channels + 1 ground)
2. USB (mini-B) port.
3. Status LED

SQ series device drivers

Windows OS SQ Series device drivers are included and should be automatically installed with ScanaStudio software. Because certain system configurations may require manual installation, the following sections provide guide lines to perform driver install or update for SQ device.

Manual driver setup on Windows To setup the drivers manually, follow the following steps:

1. Download the driver package from this url: https://cdn.ikalogic.com/dist/drivers/ScanaStudioDriver.zip
2. Unzip the driver package to a known folder, (e.g. `c:\ScanaStudioDriver\`)
3. Connect the SQ device to a USB port on your computer (**Important**: avoid using any HUBs, connect directly to root USB port.)
4. Open device manager and locate the SQ device in the list. The device may be named “IKALOGIC Device”, “IKALOGIC COM PORT” or “USB Serial port” (even though an SQ devices is not a serial port, until the driver is correctly installed, windows wont be able to correctly enumerate it). You may also disconnect and reconnect the SQ device to see which device appears and disappears

in the list. It's also worth noting that if the SQ device driver is not correctly installed, a yellow exclamation mark should be visible next to the device name in the list
5. Once the device is identified, right click on it and select "Update driver" (or an equivalent item depending on the language of your system).
6. Follow the steps in order to specify the location of the driver on your computer.
7. Provide the path to the drivers folder that was created in step 2.
8. Follow the lasts steps in the wizard to finish installing the driver.

If the device is still not recognized at this point, we recommend installing the FTDI D2XX drivers that can be downloaded here: https://www.ftdichip.com/Drivers/D2XX.htm

Linux and MacOS On newly purchased devices, Linux and MacOS do not need any specific operations to install SQ device on the system. It should be automatically recognized by the system.

In case of doubt, or if any diagnostic is needed, the SQ device uses the following USB VID / PID pair:

- VID: 0x0403
- PID: 0x7FD0

SQ devices uses an FTDI USB chip, loaded with a proprietary PID that is specific to Ikalogic products. This prevents FTDI drivers from claiming the device as an FTDI serial COM port.

For older devices (manufactured prior to 2019), the loaded PID was not a proprietary one, and was shared with other FTDI Chip devices. Luckily, it is possible to update the PID of an older SQ device by launching the device compatibility wizard (which can be accessed from the top right settings menu in ScanaStudio).

SQ LED indications

SQ status led can either be Off, Green or Orange.

Led state	Meaning
Off	Device not powered (not connected to a USB port)
Orange	Either the device was just plugged in USB but not recognized by software, or device is in generator mode
Green	Device is recognized by ScanaStudio software and operational

Introduction to ScanaPLUS logic analyzers

Important Note: ScanaPLUS is a discontinued device and have been superseded by SP209 series logic analyzers. We - of course - continue to provide full support for the users who have purchased ScanaPLUS devices.

What is ScanaPLUS?

ScanaPLUS is a 9 channel logic analyzer with 100MHz sampling rate. ScanaPLUS compresses and stream samples to the host computer.

Figure 16: ScanaPLUS logic analyzer device

Principle of operation

ScanaPLUS device mode of operation is illustrated in the image below: it samples the logic signals on all 9 channels at a maximum sampling frequency of 100 MHz, and records the exact time of each and every transition. In the same time, it compresses this data and streams it to a host computer running ScanaStudio. That implies that there is no need to adjust the sampling frequency and number of samples to reach a compromise between sampling time and accuracy: With ScanaPLUS you can use the maximum sampling frequency without sacrificing the memory usage.

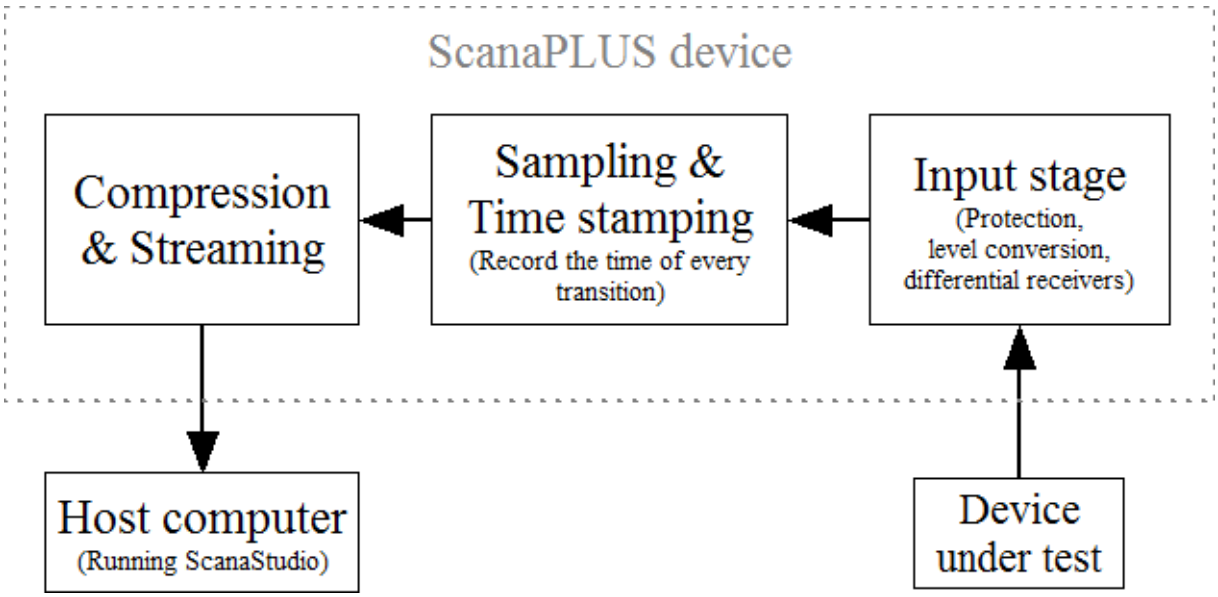


Figure 17: ScanaPLUS principle of operation

ScanaPLUS device drivers

Windows OS ScanaPLUS device drivers are included and should be automatically installed with ScanaStudio software. Because certain system configurations may require manual installation, the following sections provide guide lines to perform driver install or update.

Manual driver setup on Windows To setup the drivers manually, follow the following steps:

1. Download the driver package from this url: https://cdn.ikalogic.com/dist/drivers/ScanaStudioDriver.zip
2. Unzip the driver package to a known folder, (e.g. `c:\ScanaStudioDriver\`)
3. Connect the device to a USB port on your computer (**Important**: avoid using any HUBs, connect directly to root USB port.)
4. Open device manager and locate the device in the list. The device may be named “IKALOGIC Device”, “IKALOGIC COM PORT” or “USB Serial port” (even though a ScanaPLUS devices is not a serial port, until the driver is correctly installed, windows wont be able to correctly enumerate it). You may also disconnect and reconnect the device to see which device appears and disappears in the list. It’s also worth noting that if the device driver is not correctly installed, a yellow exclamation mark should be visible next to the device name in the list.
5. Once the device is identified, right click on it and select “Update driver” (or an equivalent item depending on the language of your system).
6. Follow the steps in order to specify the location of the driver on your computer.
7. Provide the path to the drivers folder that was created in step 2.

8. Follow the lasts steps in the wizard to finish installing the driver.

If the device is still not recognized at this point, we recommend installing the FTDI D2XX drivers that can be downloaded here: https://www.ftdichip.com/Drivers/D2XX.htm

Linux and MacOS ScanaPLUS devices uses an FTDI USB chip, loaded with a default FTDI PID/VID. A Proprietary PID was latter attributed to Ikalogic products: this prevents FTDI drivers from claiming the device as an FTDI serial COM port.

For older devices (like the ScanaPLUS), the loaded PID was not a proprietary one, and is shared with other FTDI Chip devices. Luckily, it is possible to update the PID of an older SQ device by launching the device compatibility wizard (which can be accessed from the top right settings menu in ScanaStudio software).

ScanaPLUS LED indications

Figure 18: ScanaPLUS LEDs

ScanaPLUS has two leds: 1: Power LED 2: Activity LED

The Power LED (1) - as the name implies - is tuned on when the ScanaPLUS device is correctly connected to USB port and powered up.

The activity LED (2) blinks whenever there is a logic state change that is detected on any of the 9 input channels.

Introduction to Scanalogic-2 logic analyzers

Important Note: Scanalogic-2 is the very first product launched by Ikalogic and is now discontinued. It have been superseded by SQ series logic analyzers in 2014. We of course continue to provide full support for the users who have purchased Scanalogic-2 devices.

What is Scanalogic-2

ScanaLogic-2 is a compact and portable logic analyzer and signal generator. It directly connects to a computer via an USB connection. The PC given software is used to display the signals and the decoding, generate signals and configure capture parameters (like trigger option, sampling frequency, etc.). It's very simple to use, it is easily mastered by beginners and professionals, thanks to the intuitive PC software featuring all the scrolling and navigation options you would expect in any professional software. Scanalogic-2 introduces new, very useful features, like the mixed mode, allowing you to generate/playback digital signals, while recording the response on other channels, all at the same time!

Figure 19: scanalogic-2 logic analyzer

Principle of operation

ScanaLogic-2 relies on a set of embedded RAM memory chips that store samples on 4 channels simultaneously. The maximum sampling frequency is 20MHz and the RAM is able to store 256K points per channel. It's possible to reduce the sampling rate in order to increase total capture time.

Input/output specifications

- Input voltage range: -0.7 to 5.5V
- Compatible logic levels: 2.8V, 3.3V, 3.6V and 5V.
- Output voltage level (fixed): 3.6V.
- Fastest digital signal that can be captured or generated: 2 MHz

Mixed mode (capture & generate) Scanalogic-2 allows signals to be generated on 2 channels, while signals to be captured on the 2 other channels. Unlike newer SQ series devices, Scanalogic-2's output is fixed to 3.6V push-pull. (SQ series offer much more versatile probes I/O configuration).

Note: When operating Scanalogic-2 in mixed mode (Capture & generate), it is only possible to set two channels as "output", while the two others are "inputs". It's not possible, for example, to set 3 channels as outputs and 1 channel as input. (This limitation have been addressed in newer SQ series logic analyzer and pattern generators).

Status LED

The ScanaLogic-2 device has one yellow status LED, which can indicate different statuses:

- LED Blinking one time per second : Device is in stand by mode. All probes are in HI-Z (high impedance) mode.
- LED Blinking very quickly : Device is transferring data from or to the host computer over USB.
- LED ON continuously : Device is in generator mode

Software

Just like all other Ikalogic logic analyzers, ScanaStudio software is needed to configure the logic analyzer, to download captured signals and to visualize them.

Currently, Scanalogic-2 is not supported by latest ScanaStudio releases (3.0) however, it's planned to reintroduce support for Scanalogic-2 devices in future software releases.

For the time being, to use a ScanaLogic-2 device, you can download an older ScanaStudio version (V2.3) from the links below:

- Download for windows
- Download for Linux (32)
- Download for Linux (64)
- Download for MacOS

Drivers

Scanalogic-2 enumerates on the USB port as an HID device (Human Interface Device). Therefore, no any driver is needed on any of the supported platforms (Win/Mac/Linux). It is automatically recognized just like a keyboard or a mouse.

Scanalogic-2 Edu Kit

A low cost variant of this product is the Scanalogic-2 Edu-Kit (Educational Kit)

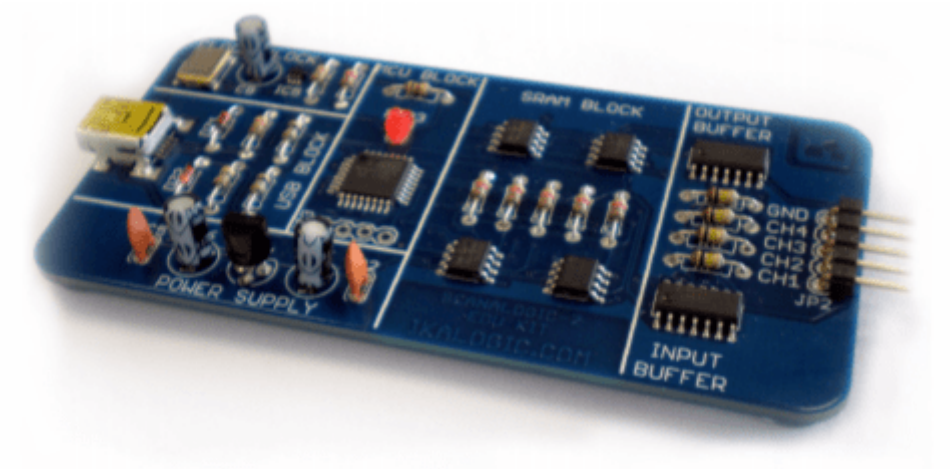


Figure 20: Scanalogic-2 educational kit

Once fully mounted, this DIY (Do It Yourself) kit, will provide the same function as the standard Scanalogic-2 device.

The kit assembly instructions can be downloaded here:

http://cdn.ikalogic.com/docs/sl2-edu-kit-guide.pdf

ScanaStudio user interface

ScanaStudio user interface is designed to be very simple and intuitive. All main features are accessible from the main window. It's important to familiarize with the few different menus and zones of the application to be able to easily follow the rest of this manual.

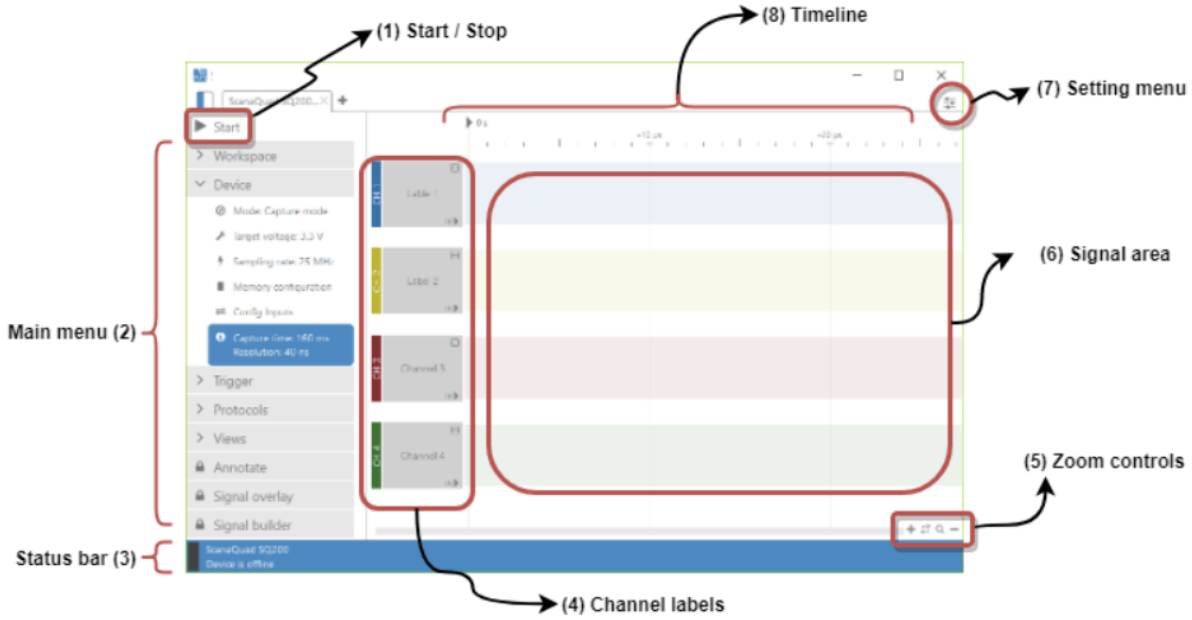


Figure 21: ScanaStudio user interface

The main menu (2) is the most important part of ScanaStudio interface. It's organized in tabs, grouping different features in categories.

The status bar (3) shows information related to the device and to the last capture. For instance, the sampling rate associated to the signals being displayed on screen can be found in the status bar.

The settings menu (7) can be used to change application settings like color themes, keyboard shortcuts and software update parameters.

Creating workspaces

What is a workspace? In ScanaStudio, a workspace is an entity that is closely related to a Logic Analyzer device. A workspace stores the captured samples and all the configuration related to that device, like sampling rate or sampling depth (number of samples, or total capture time). A workspace will also store markers, trigger configuration, measurements, protocol decoders, buses and any other configuration made by the user. A workspace can be saved and restored later, which may be useful to recover complex configuration and/or captured samples.

To create a new workspace, simply press CTRL+N (unless you have modified the default shortcut in the setting menu). You may also create a new workspace by going to Main menu > Workspace > New.

Depending on which device is connected to your computer, a new workspace dialog will be displayed, prompting you to select a device to continue.

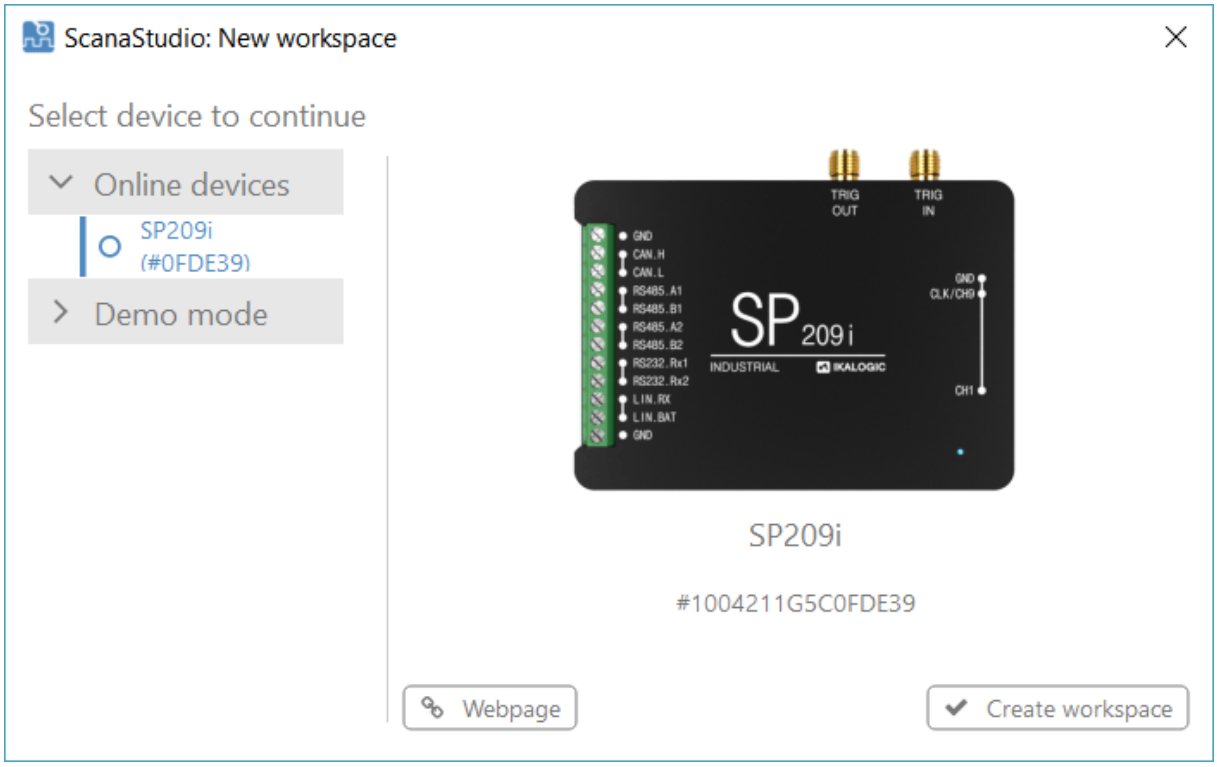


Figure 22: Creating new workspace

You may select any device (even if it's not connected to your computer), under the "Demo" menu in the New workspace dialog. In that scenario, a workspace is created in demo mode (which will be explained in next sections). Once a compatible device is connected to a demo workspace, it can be converted to a "normal" workspace. This means you can start configuring a workspace in demo mode, to prepare a complex measurement (for example) and connect a logic analyzer later (provided it is the same logic analyzer device as the one selected when creating the demo workspace).

Demo mode operation In demo mode operation, ScanaStudio will operate as if a real device was connected to the computer. When a user tries to capture logic signals in demo mode operation, pseudo-random signals will be generated. Some protocol decoders (that have the demo signal generation capability) can even generate demonstration signals to allow the user to test the protocol decoding feature.

The demo mode can be used for the following purposes:

- Testing the functionalities of a logic analyzer before purchasing one (test driving).
- Setting up a complex workspace with protocol decoders in advance, before performing actual captures.
- Generating known and repeatable logic patterns to test a protocol decoder. When writing a new script to decode a protocol, it's actually recommended to start by writing the routines to generate logic signals for that protocol. This way, the script author can generate and test as many patterns as he deems necessary.

Starting a new capture

Before starting a new capture, you first have to create a workspace for the logic analyzer device you wish to use.

Then, you need to configure the logic analyzer device by going to the "device" tab in the main menu. Device configuration will be handled in detail in next chapter. Default configuration is usually well adapted for a generic logic capture, thus, default parameters can be kept by default if you're attempting a first capture and you're not certain about what are the best parameters to be used.

Depending on the signals that need to be captured, you may need to setup a trigger by going to the "Trigger" tab in the main menu. Trigger is a major topic and will be discussed in detail in next chapter. Without any trigger setup, capture will simply start as soon as the "Start" button is clicked.

Finally, Click the "Start" button to start the capture. As soon as the capture is started, you should see signals appearing in the signals area.

Navigating trough captured samples

Once signals appear on the screen, one can start navigating through captured signals. By "navigating", we mean operations like zooming in and out or panning right and left. Depending on the number of channels, measurements and virtual channels (like buses) that are displayed, a user may also need to pan up and down to see available signals.

Navigation using a mouse The most convenient way of navigating through the samples is by using a mouse equipped with a mouse wheel.

- You can zoom in and out using the mouse wheel. Pressing the ALT key while turning the mouse wheel allows you to zoom in fine steps.
- You can pan right and left by clicking and holding the mouse left button, then moving the mouse right and left; the signal area will follow the mouse.

Navigation using a mouse pad All mouse pads are different depending on OS and manufacturer, but exact same functions can be accomplished as with a mouse.

Navigation using a touch screen Laptops and computers that have touchscreen support are also supported by ScanaStudio, that is, you can do pinch and stretch gestures to zoom in and out. You can also slide the signal area to pan right or left.

Navigation using the keyboard Using a keyboard, one may pan right and left using the right and left arrows in a keyboard. It's also possible to zoom in and out using the shortcut keys defined in the settings menu and may be different on Windows, Mac and Linux.

More navigation tips and tricks It's possible to quickly jump to the next transition is a signal by placing the signal at the extreme right or extreme left of a signal until this arrow appears:

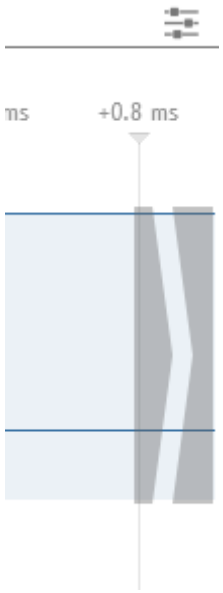


Figure 23: Navigation to next transition

Clicking this arrow will let you navigate to the next transition, without changing your zoom settings.

The icons at the bottom right of the signals are can be used to: Zoom in, zoom to fit, zoom to region and zoom out (in that order).

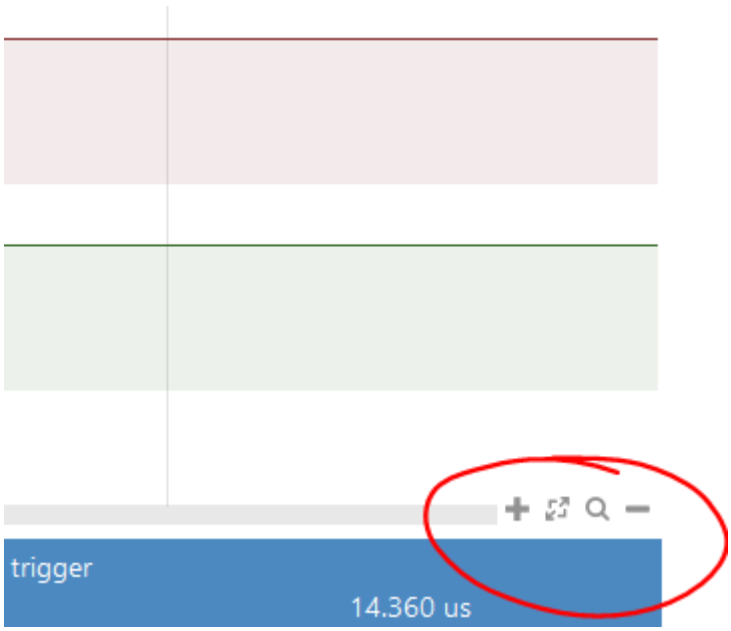


Figure 24: navigation tools

Annotations and measurements

In ScanaStudio, it's possible to annotate a capture by adding markers to a workspace. A marker is a vertical line that is attached to a certain time in the capture. Another kind of annotations are measurement, which can be used to measure the frequency or the duty cycle or count the number of edges (the list is not exhaustive). A measurement is delimited by two markers, thus, adding a measurement will forcibly add two markers to the workspace (unless the measurement is added on pre-existing markers).

Below is an example measurement constrained by markers 1 and 2.

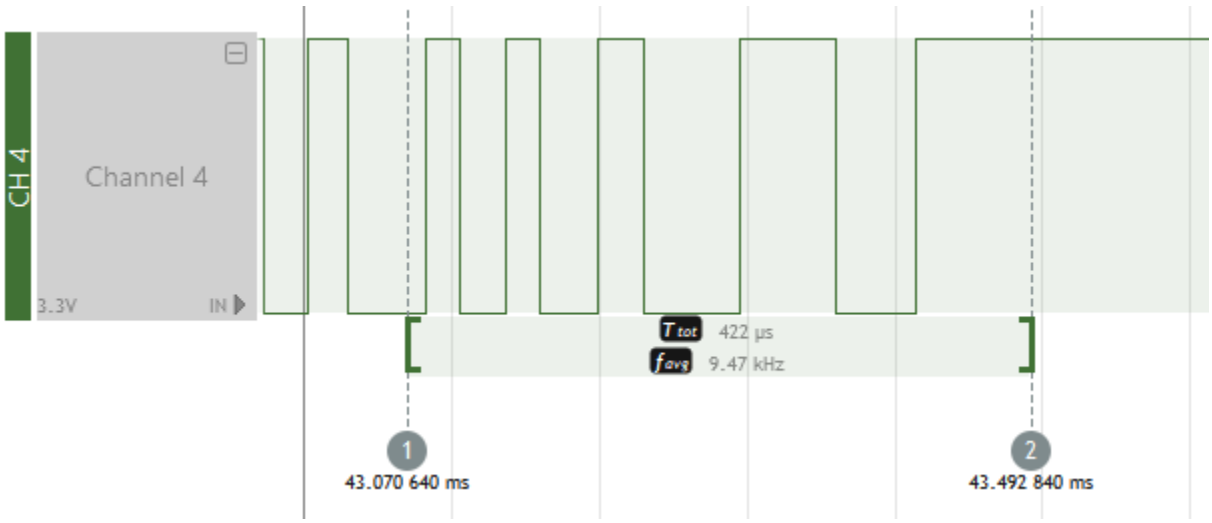


Figure 25: Example measurement

Standard measurement vs custom measurement When working with measurements in ScanaStudio, you'll notice there are two kinds of measurements: Standard and custom. The only difference between the two is the number of measures being made. By default, standard measurement will include time and frequency. Adding custom measurement will open a dialog letting you chose among many other measures like:

- Total time
- Total logic-high or logic-low time
- Average frequency
- Minimum and maximum frequency
- Average duty cycle
- maximum and minimum duty cycle
- Falling and rising edges count

When adding a custom measurement, you can check the option "Make this the default measurement configuration" as shown in the image below to make it the "standard" measurement.

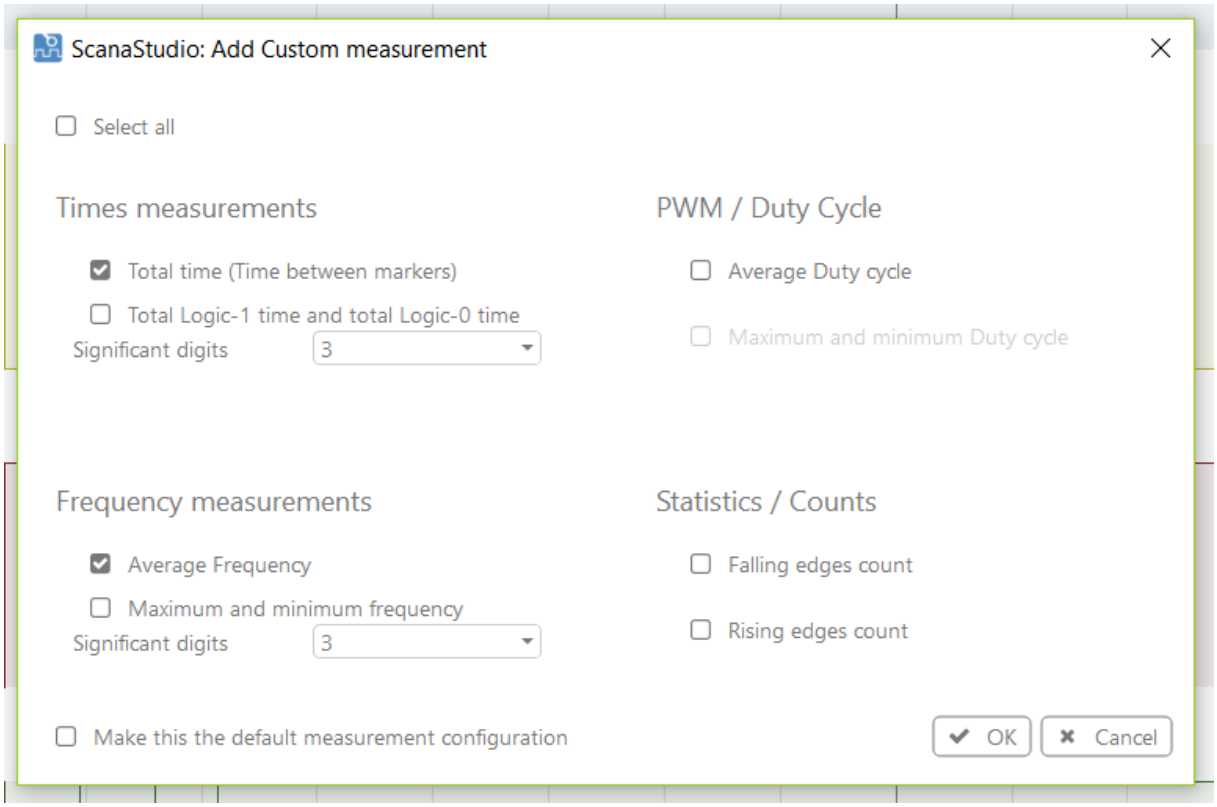


Figure 26: Custom measurement

Adding a marker To add a marker, you can right click on the signal area and select “add marker” as in the image below:

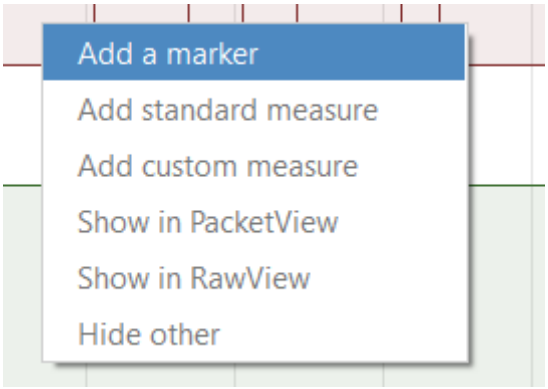


Figure 27: Add marker

You can also click “Add marker” in the annotate tab in the main menu:

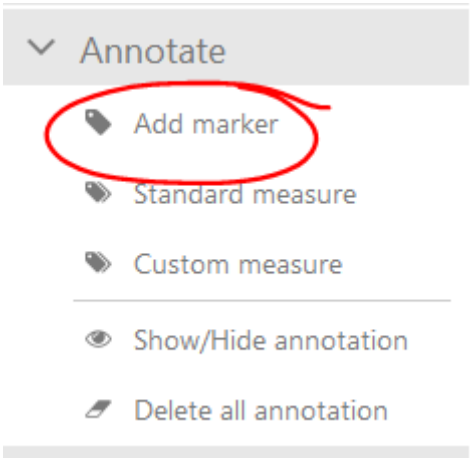


Figure 28: Adding a marker in annotate menu

You'll notice that a marker will be displayed under the mouse cursor. Move the mouse cursor at the position where the marker need to placed, then left click to fix the marker position.

You'll notice that there is magnet effect between the markers and the signal edges. This is usually helpful to align marker on signal edges, however, if this feature is not wanted, it can be inhibited by clicking on the ALT key on your keyboard.

Trigger markers Depending on the trigger configuration, a special trigger marker may be added to the workspace. As the name implies, this is a special marker that marks the exact trigger instant. A trigger marker may not be moved or deleted.

Moving an existing marker To move a marker, simply follow those steps:

1. Click once on the marker with the mouse (or mouse pad). You'll notice that the marker will follow the position of the mouse cursor
2. Move the mouse cursor (and the marker) to the new position
3. Click a second time on the signals are to fix marker's new position.

Adding or editing a measurement To add a measurement, simply right on the signal area and click "add measurement". You can also click on the "standard measure" in the annotate tab in the main menu.

When adding a measure, you will be required to add two markers defining the start and the end of a measure. Simply follow the steps required to add the two markers to finish adding a measurement.

Note: You may use existing markers instead of adding new ones. To do so, simply place the new marker above an existing one: the existing marker will be used to delimit the measurement and no new marker will be added.

You can always right click on a measure and select “customize this measure” to change the list of parameters being measured.

Moving a measurement You can right click on a measurement and select “Move measurement” as in the image below.

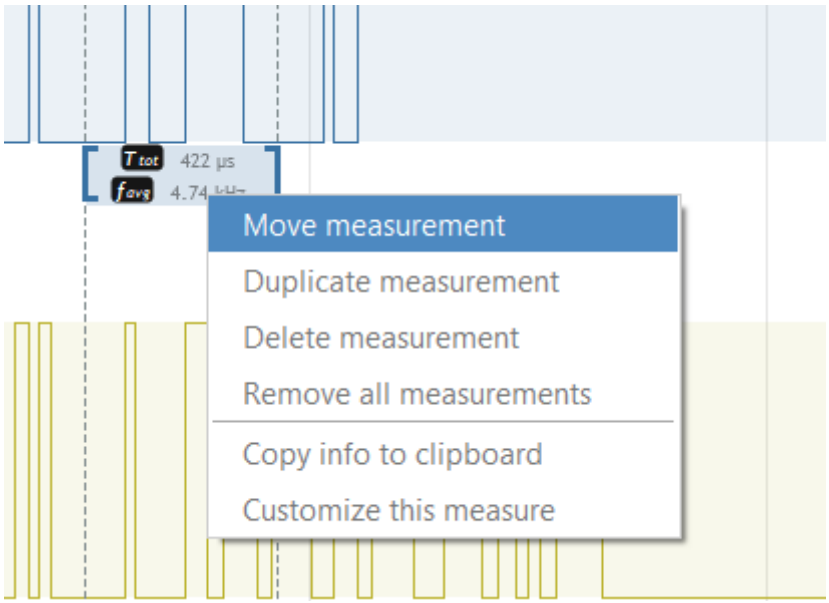


Figure 29: Moving a measurement

When a measurement is being moved, it will simply follow the mouse’s cursor. Select the new position for the measurement and left-click with the mouse to fix the measurement in it’s new position.

It’s also possible to precisely adjust the left and right boundaries of a measurement by zooming on each marker and moving it (just as you would move any regular marker).

ScanaStudio options menu

The options menu can be invoked by clicking on the settings icon as shown in the image below:

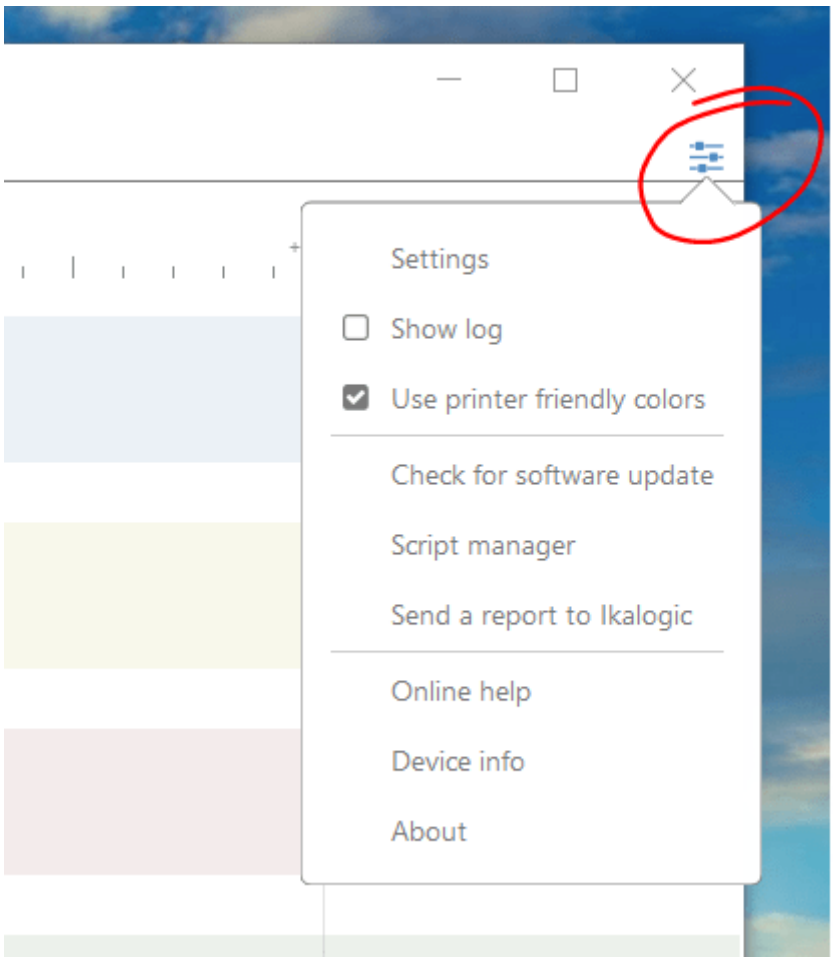


Figure 30: settings menu

This menu lets you access general features and configuration options of ScanaStudio. Here is a brief descriptions of the options menu items:

- Settings: open the general settings window
- Show log: Shows the script log that may show script debugging information. Note that in case an error or warning message is emitted by a script the log will display automatically.
- Printer friendly colors: used to switch between a back and write background.
- Check software updates
- Script manager: lets you check your local library of scripts, and update it from the online repository if needed.
- Send a report to Ikalogic: Can be used to send us a detailed usage log of the application, along with a small message to help us understand the problem or issue you're facing.
- Online help: Lets you access this online guide
- Device info: shows information about the currently connected device

- About: shows general information about ScanaStudio like the version and authors

Capturing logic signals

In this chapter, we're going to focus on how to simply capture some logic signals using an Ikalogic logic analyzer. Please note that for the sake of clarity, we're not going to get into advanced features like configuring industrial inputs (like the ones that come with the SP209i or setting up complex triggers). Below we're going to present a recommended approach when apprehending a new measurement. You may very well have you're own (even better) techniques, so feel free to follow different approaches if you feel confident enough.

Step 1: setup the workspace

Connect your logic analyzer to your computer, and ensure it's correctly installed. Start by creating a new workspace (simply hit CTRL+N, or click "create new workspace" button):

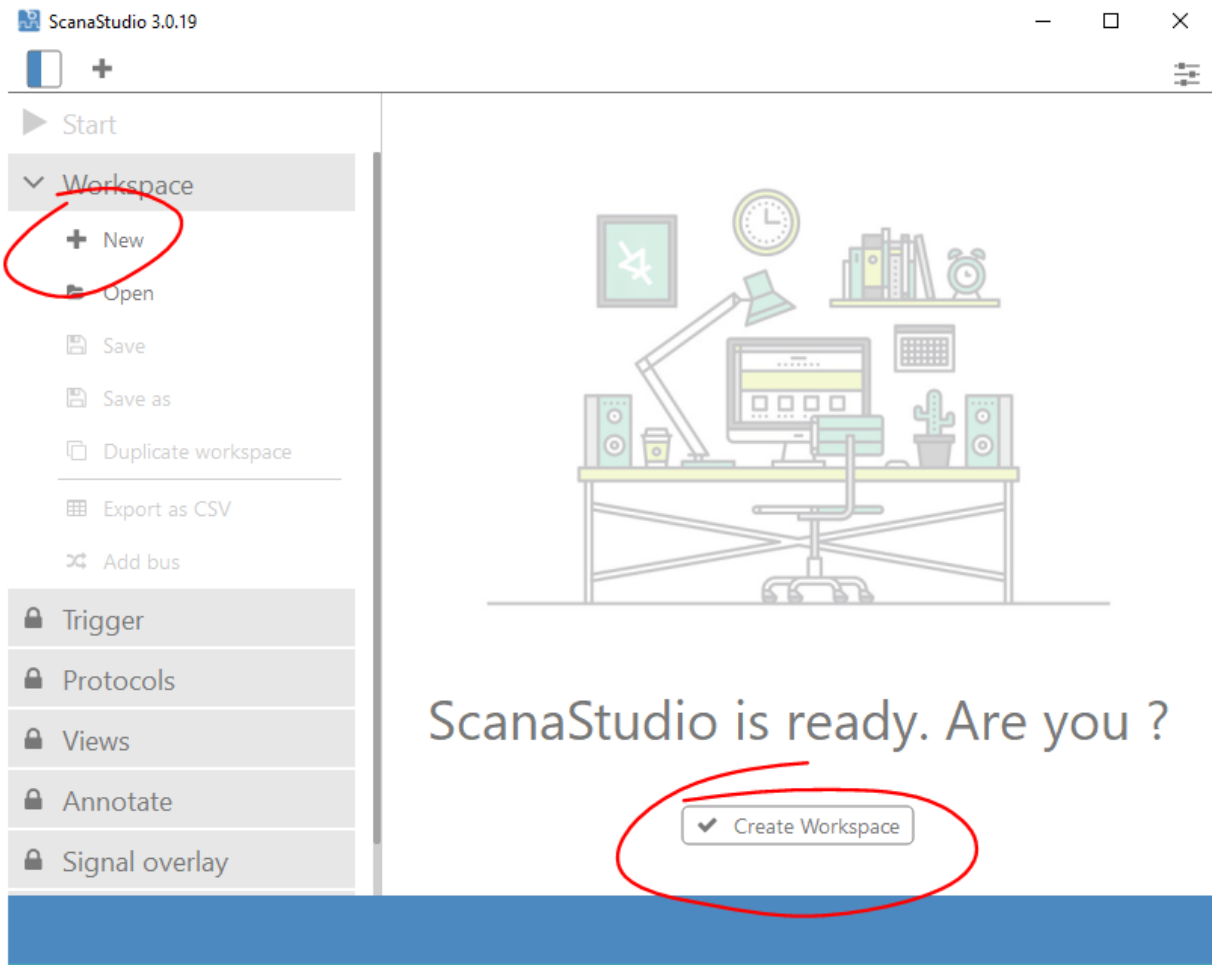


Figure 31: create new workspace

Select the right device from the list and create the workspace. If the device is correctly recognized by your computer, you should see the status indicator showing the device as “online” as shown in the image below:

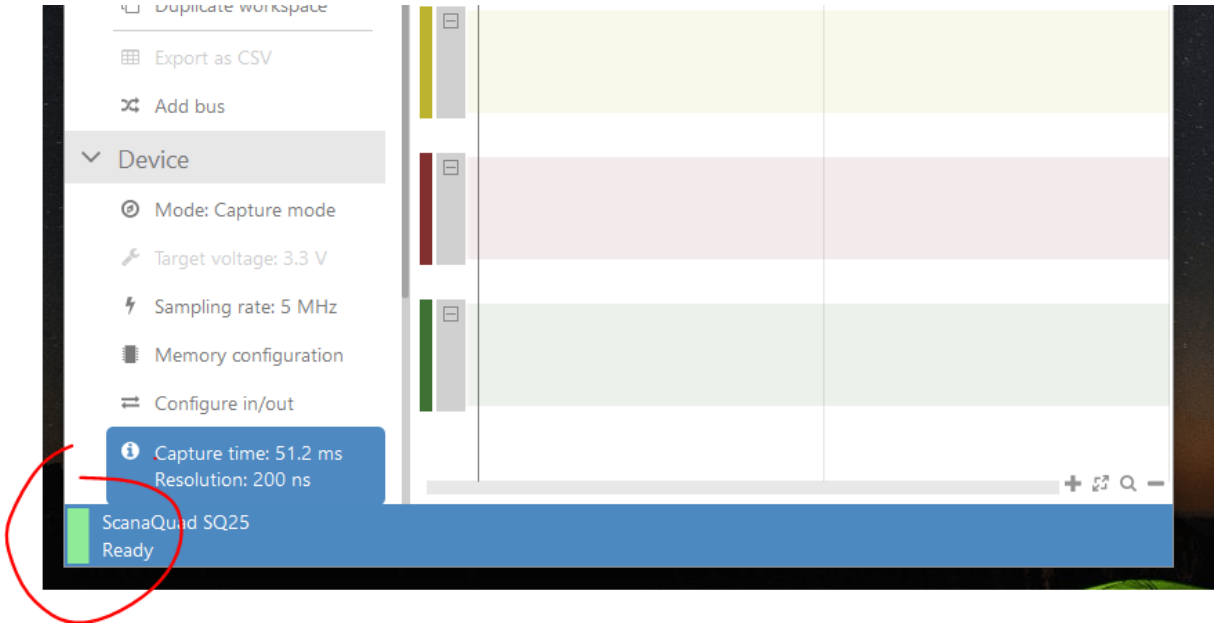


Figure 32: SQ25 logic analyzer online

Step 2: Electrical connections

If your logic analyzer’s probes are not connected to the board under test, go ahead and connect it.

- Before making any connection, ensure the voltage levels are within the operating range of the logic analyzer device.
- Also ensure that the device under test is not connected to the mains and that there are no risks of ground loops, as this may be destructive for your board, your logic analyzer or in a worst-case scenario, your host computer to which your logic analyzer is connected.
- Don’t forget to connect the ground of the logic analyzer to the ground of the board / device. **A good - noise free - ground connection can make the difference between glitch-free captured signals and noisy captures.**
- Ensure your board/device is turned on.

Step 3: First capture

Now, we recommend capturing a first set of logic signals to get a general overview of the logic levels on the different channels and ensure all signals look as expected. The image below shows an example

capture:

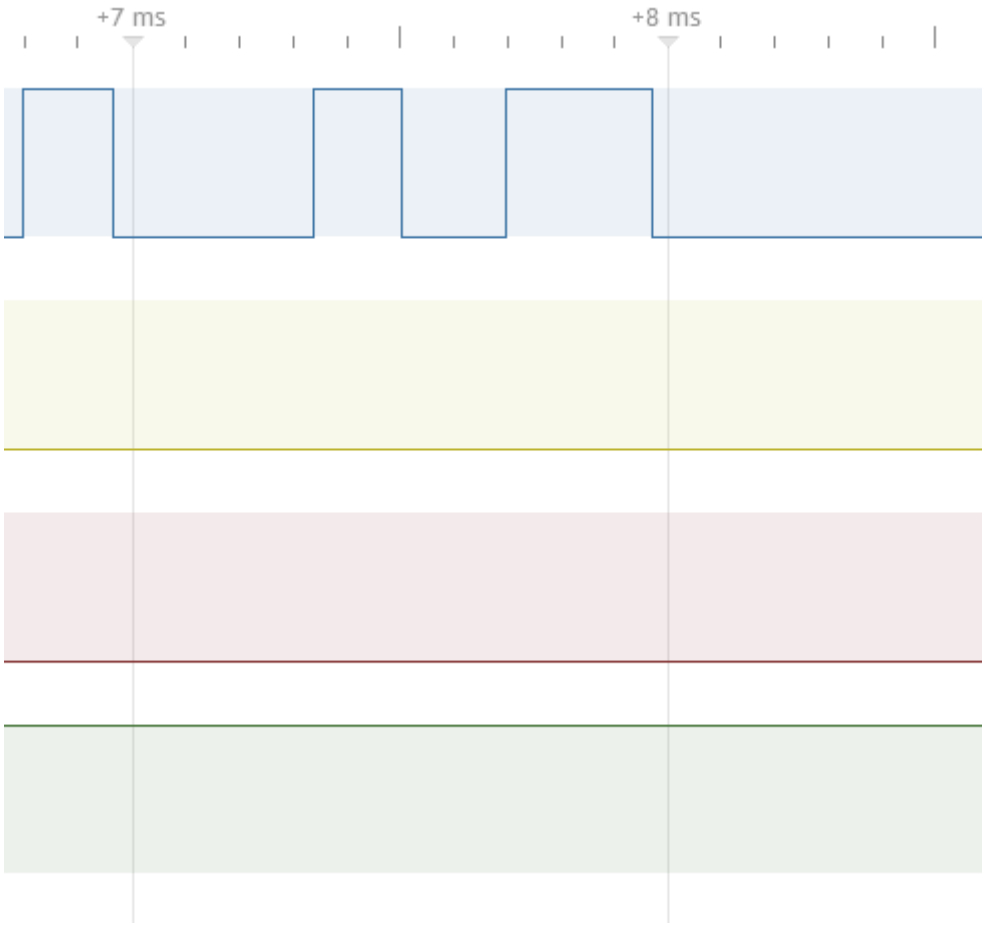


Figure 33: example logic signals

Please note that this totally depends on your board and the signals you're observing and more importantly, the logic analyzer device configuration (threshold levels and sampling rate). In your case, you may see only flat signals, which may be normal and expected.

Step 4: Further configuration

If you don't see expected signals, you may need to configure the logic levels (logic thresholds) and, if the logic analyzer device allows it, the sampling rate. You may change those parameters from the "Device" tab in ScanaStudio.

In some cases, you may need to setup a trigger to capture intermittent signals that occur only once in a while. We're not going to explain what is a trigger, but very shortly, a trigger will let your logic

analyzer device wait for a particular event before starting a capture. This configuration can be made in the “Trigger” tab, just under the device tab:

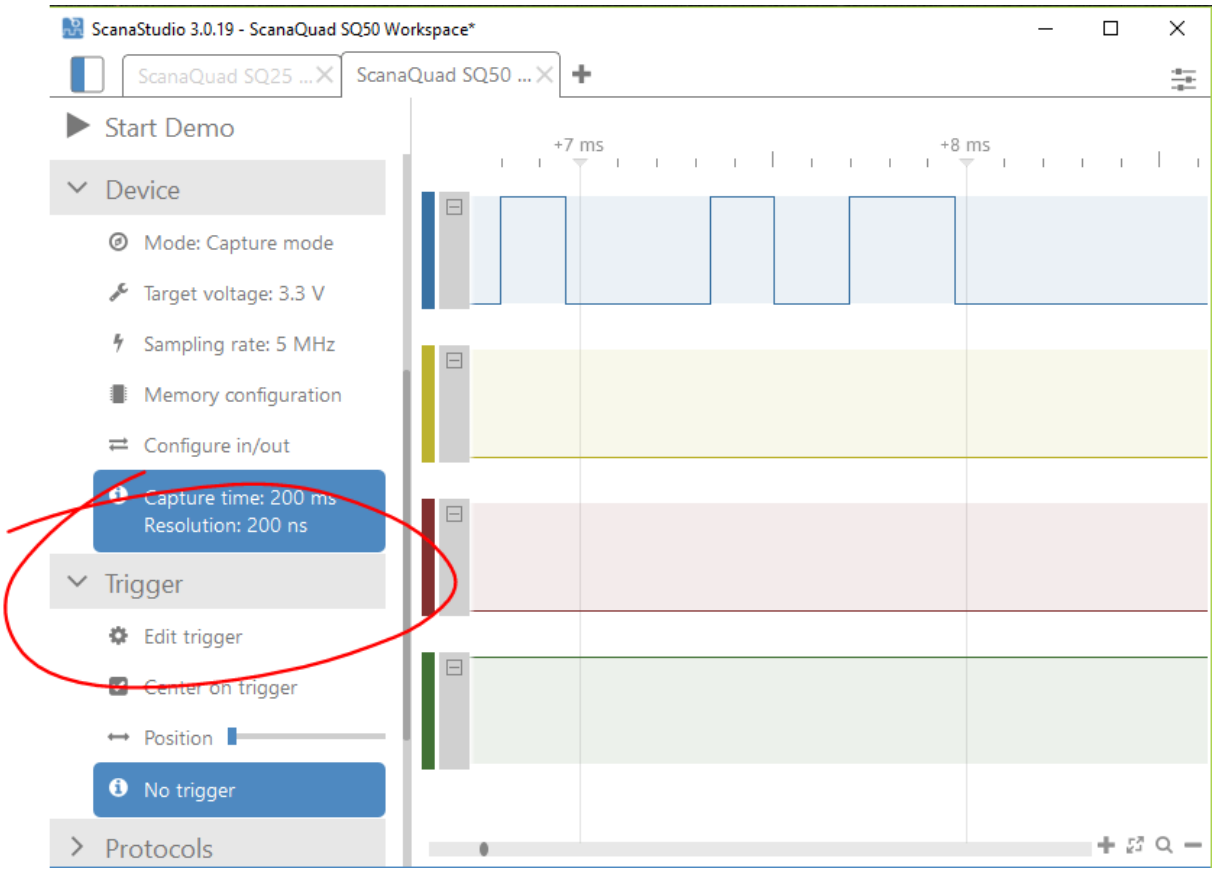


Figure 34: Trigger configuration

A very easy trigger configuration that lets your logic analyzer wait for any activity on the logic channels, is to wait for any change on any channel:

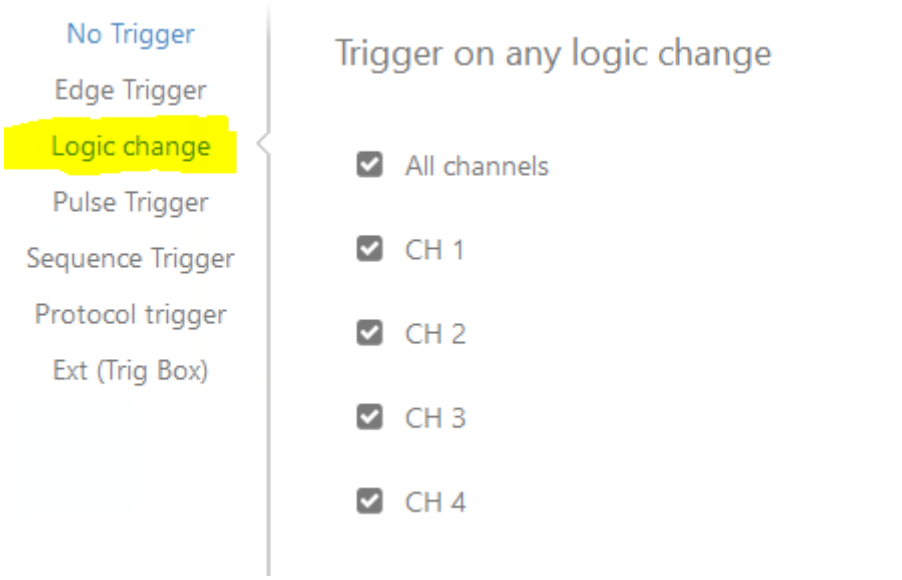


Figure 35: trigger on logic channel

Note: Both device configuration and trigger configuration will are handled in detail in other chapters.

Now that you know how to capture signals, it's time to learn how to use the full potential of ScanaStudio and your logic analyzer and decode (interpret) captured signals!

Protocol decoder

Protocol decoding is probably *the* most wanted feature in logic analyzers. Protocol decoding is the (automatic) process of analyzing the logic signals and interpreting it according to a specific protocol. For example, the image below shows I2C signals signals with an additional layer of decoding:

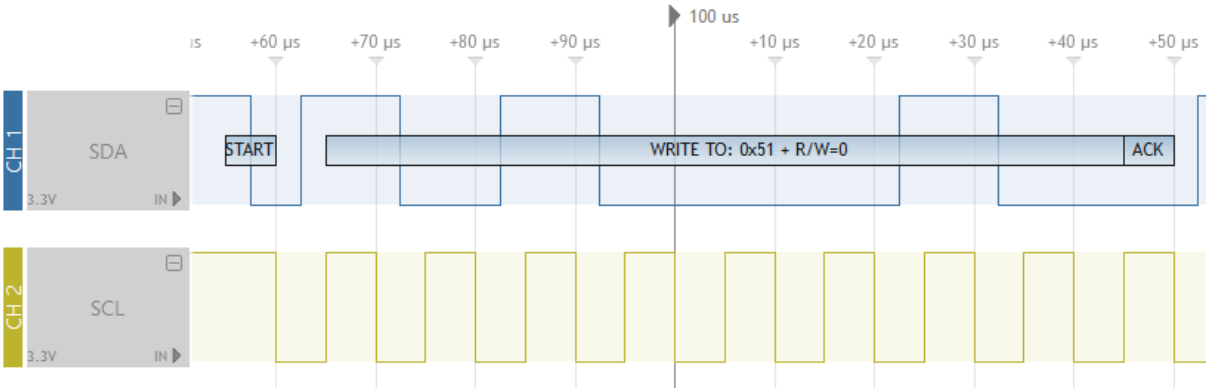


Figure 36: I2C signals decoded by logic analyzer

In the past, to verify if a protocol was correctly working, one would look at the signals with an oscilloscope, manually interpret bits and bytes, and write them down on piece of paper. Luckily, those times are over!

Knowing how protocol decoding is important to the user, we (Ikalogic) have put a great deal of effort into allowing protocol decoders report as much information as possible. For example, the CAN bus frame below has a wrong CRC, and the decoder highlights this CRC error and even displays what *should* the CRC be, according to the latest CAN standards.

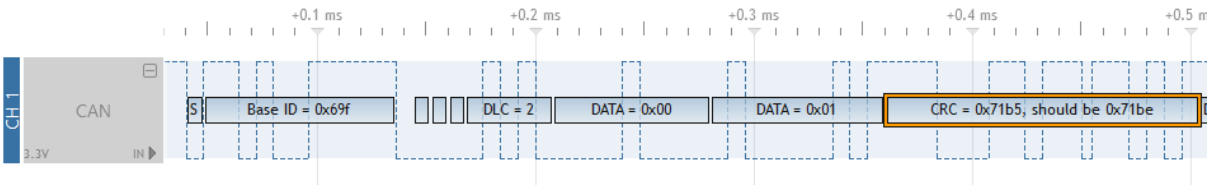


Figure 37: CAN frame with CRC error

Similarly, it will clearly show if decoded packets are correctly constructed. For instance, the diagram below shows a LIN message frame with a correct checksum value:

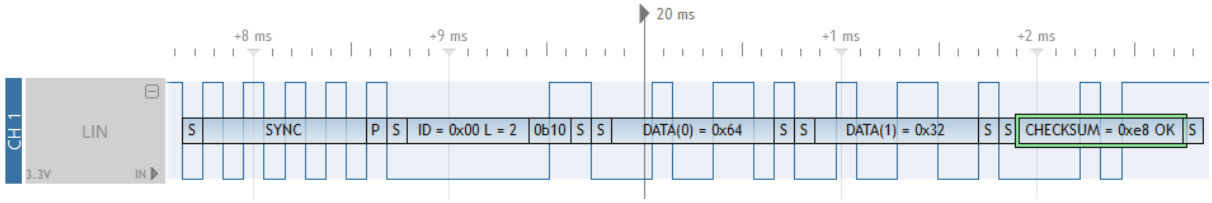


Figure 38: LIN good CRC

This implies that ScanaStudio decoders *can* calculate the CRC or Checksum of a frame and compare it against the one captured in the frame. Please note that each protocol is different, so is the what data is interpreted and presented. For instance, not all protocols specify error checking and data integrity verification like CAN and LIN bus.

- ScanaStudio scripting manual

How (and when) to add a protocol decoder?

To add a protocol to workspace, simply go to the “Protocols” tab and click “Add new” as shown in the image below:

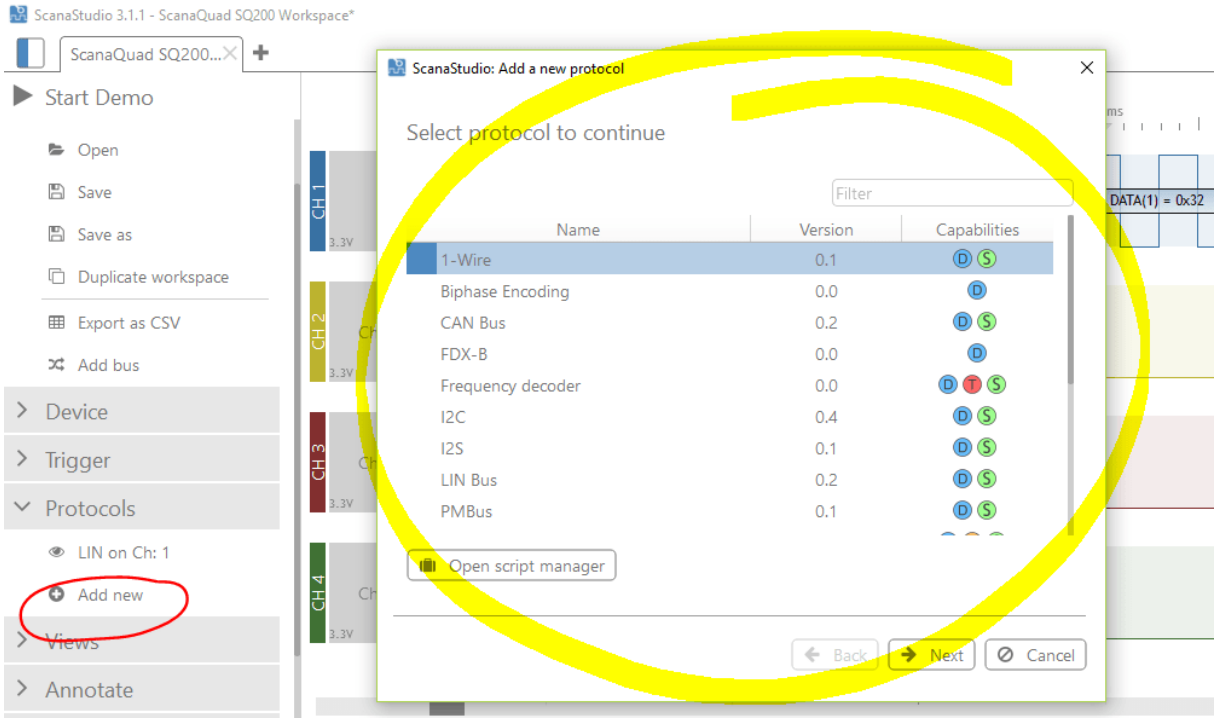


Figure 39: Add a new protocol decoder (analyzers)

Then follow the instructions to configure the decoder. Please note that every protocol is different, so is

its configuration, but it will at least imply selecting the channel(s) to be decoded. Depending on the complexity of a decoder's user interface, one may read additional documentation by clicking on the "Help" button:

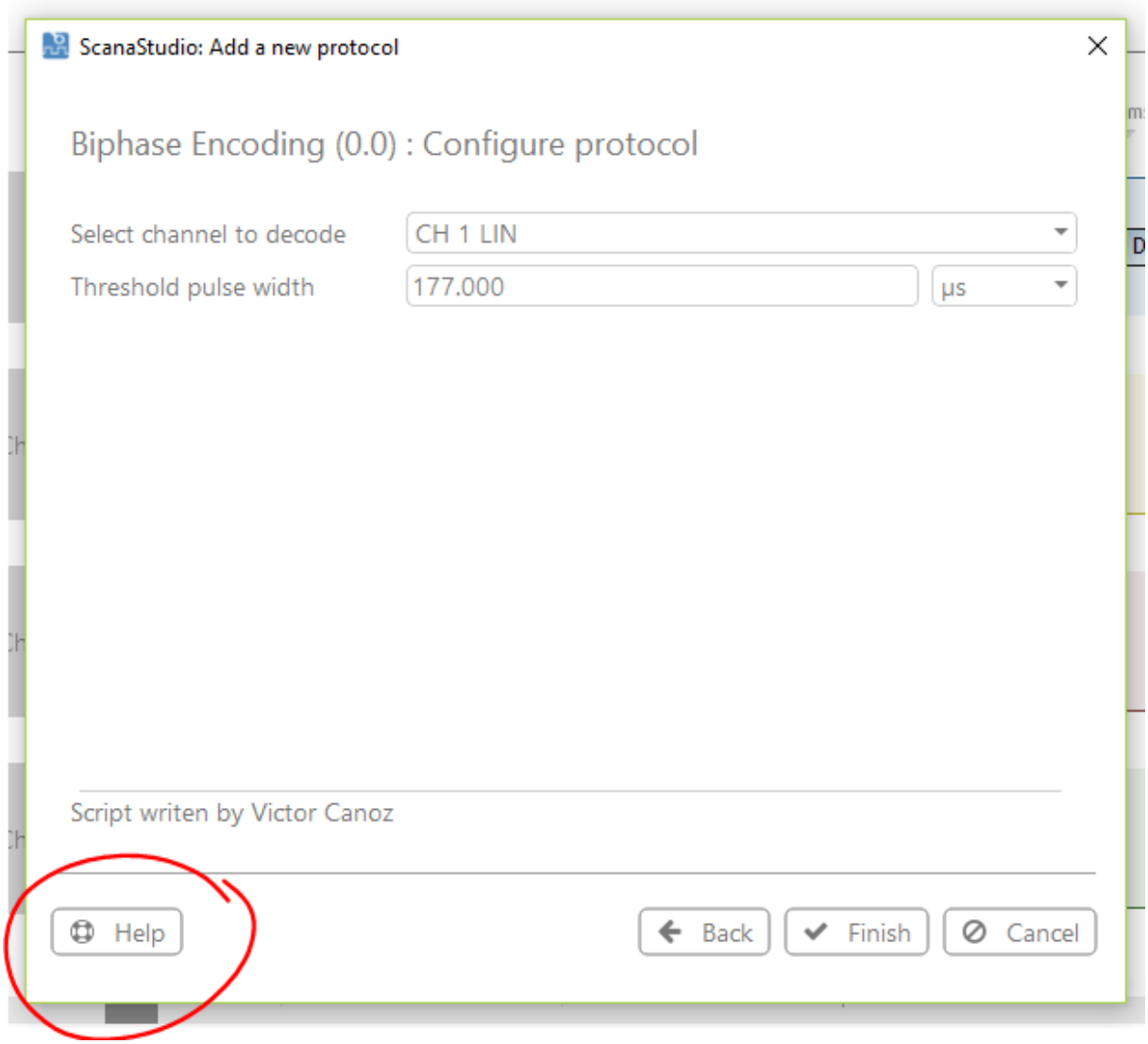


Figure 40: Protocol decoder help button

But the most important question is when to add a protocol decoder when apprehending a new system that need to be diagnosed, debugged, or simply analyzed. In an ideal world, one would start by adding a protocol to a workspace, configure it down to the smallest detail, then start a capture and voila, data would appear perfectly decoded on the screen! While this scenario may happen sometimes, things don't always go that smoothly.

We recommend to first start by capture signals, without worrying about the signal decoding part. Your

"human" eye analysis of the signals, their frequency, shape and periodicity can already give you some insights:

- You may recognize that some channels are not correctly connected or missing
- You may see that the voltage levels need some adjustments
- You may also notice that the interesting part of the signal happens only once every few seconds, so it would be a good idea to add a simple trigger.

At this point in time, you should already have enough information to add and configure a protocol to your workspace.

Managing the list of supported protocols

A note about scripts ecosystem In ScanaStudio, protocol decoders (analyzers) are part of the "Scripts ecosystem". ScanaStudio relies on scripts to add various features to the software, among which, the protocol decoding. Each protocol is supported via a script that cover one or more feature like:

- Signal decoding
- Trigger sequence generator (allowing your compatible logic analyzer to trigger on a specific protocol feature)
- Signal builder (allowing a generator-capable logic analyzer to generate specific packets for a protocol)

Scripts are open-source and highly collaborative. Ikalogic remains the main contributor to the ScanaStudio scripts, always ensuring scripts meets our standards, but we couldn't get to where we are without the input of many active users who suggested fixes, updates and new features. Some scripts are written from the ground up by other third party users and shared back to the whole community. If you're interested about adding your own features to ScanaStudio to automate some analysis processes, you can get started here:

Updating decoders and/or getting new ones At any moment, you can launch the script manager from ScanaStudio by clicking on the settings menu (top right corner) and then "Scripts manager".

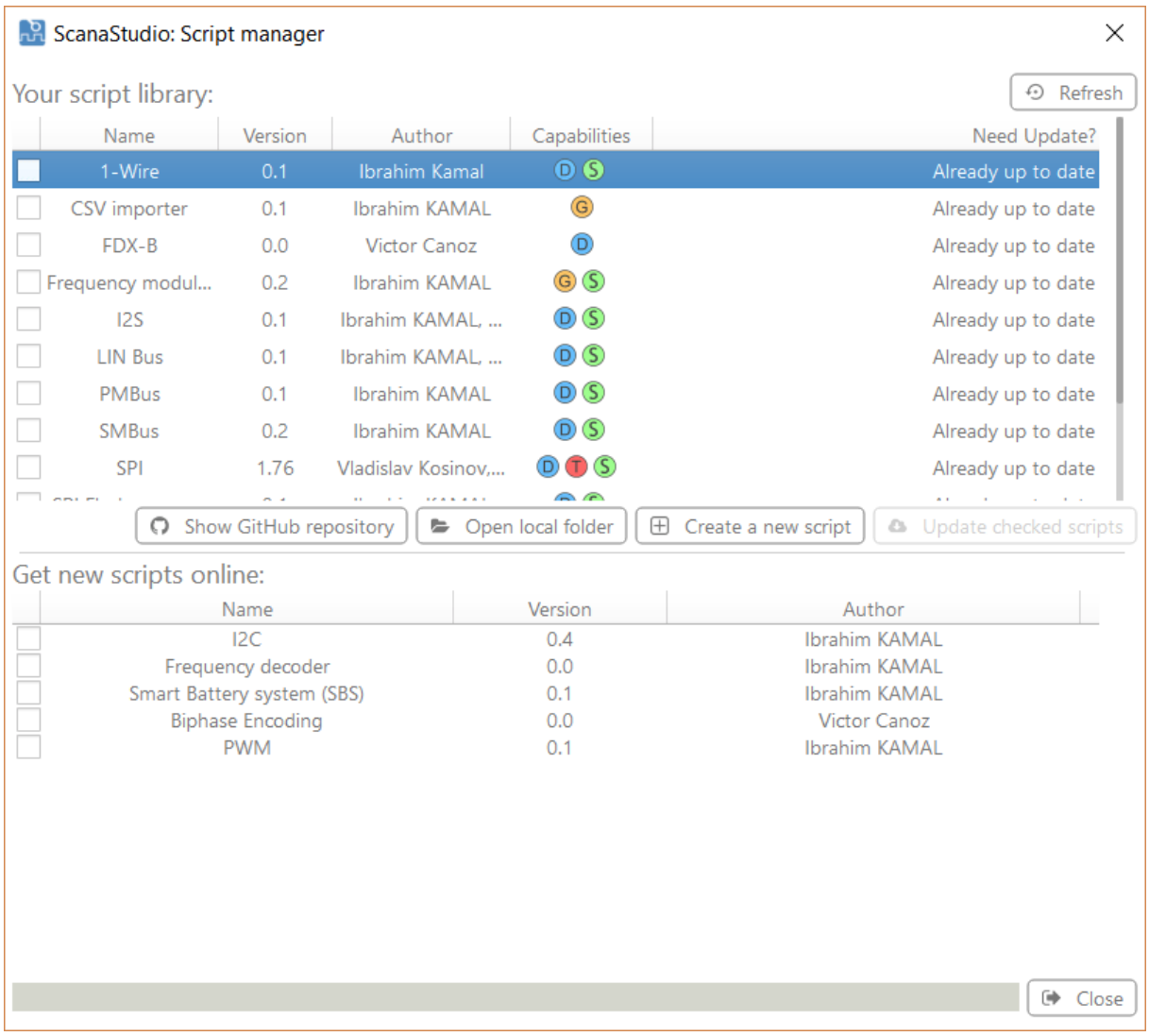


Figure 41: script list

This windows lets you download any new scripts that may have been added online, or update existing scripts. Scripts that have the “D” icon in the capability column are scripts that can decode signals.

Getting the most out of a decoder

Don’t hesitate to look through the configuration options that are specific to each protocol decoder. Most decoder can be configured to display the data in various formats (HEX, ASCII, Binary, etc). Some even allow you to choose different formats for different kind of fields. For instance, the I2C protocol analyzer lets you select a different display format for both DATA and Address fields:

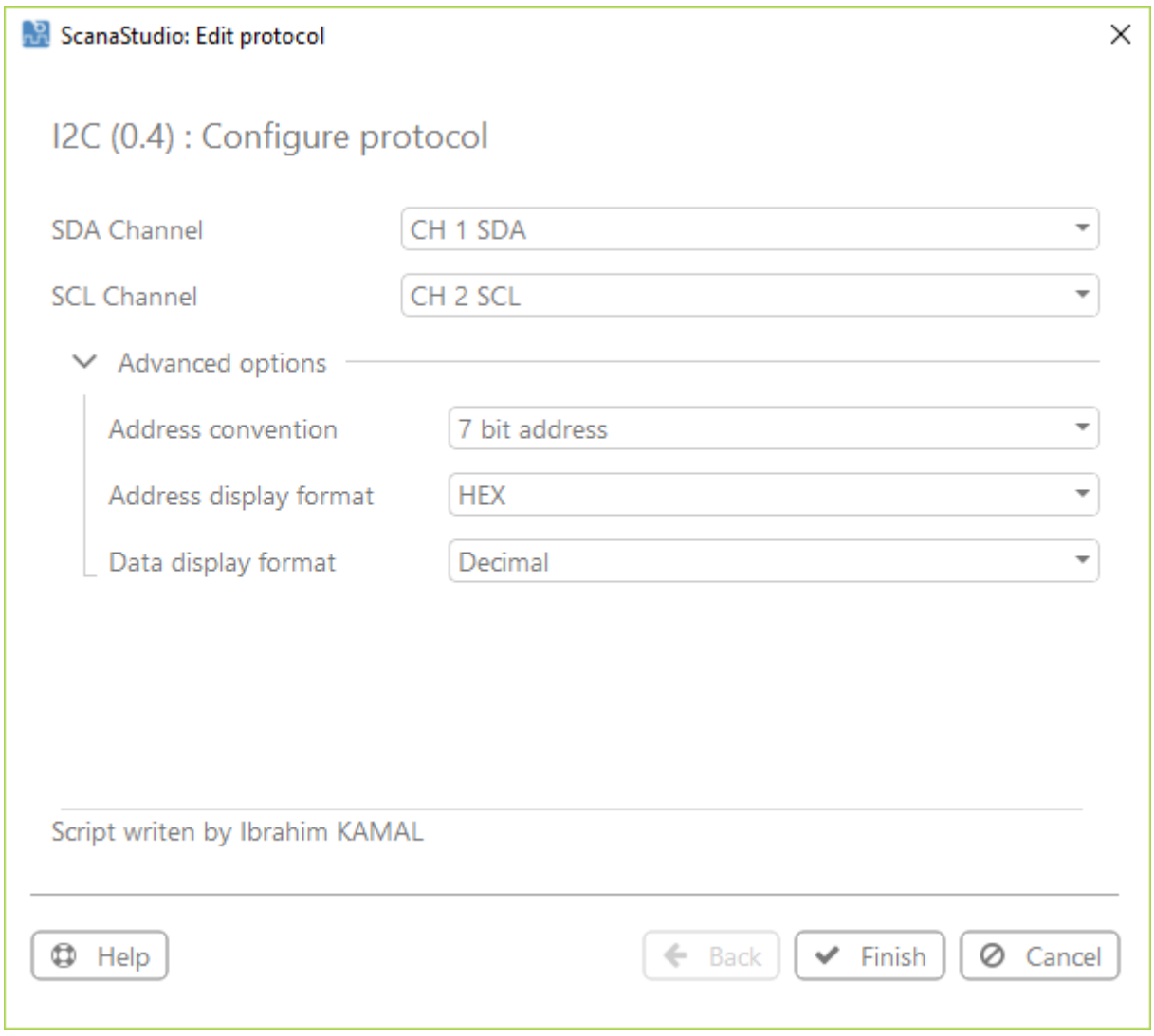


Figure 42: I2C protocol decoder display options

Which can produce this result:

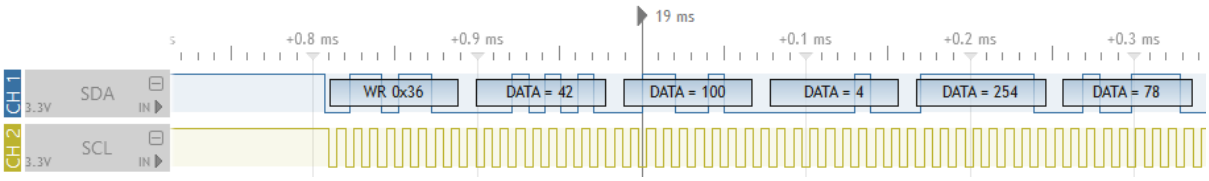


Figure 43: I2C packets analyzed

Alternative data views

Many (if not all) protocol decoder add-on scripts produce an important amount of data. Whether it's SPI, UART or CAN bus, it can be non practical to search for the data directly on the logic signals waveform (as in the image bellow):

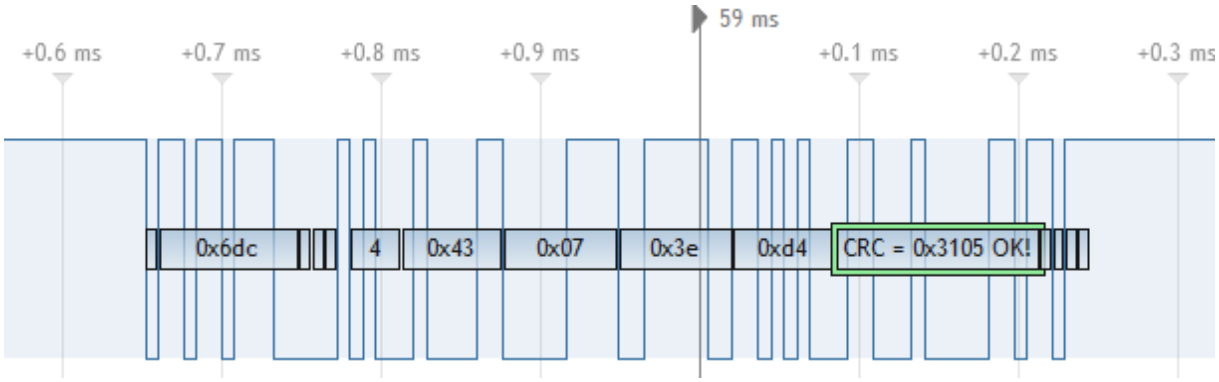
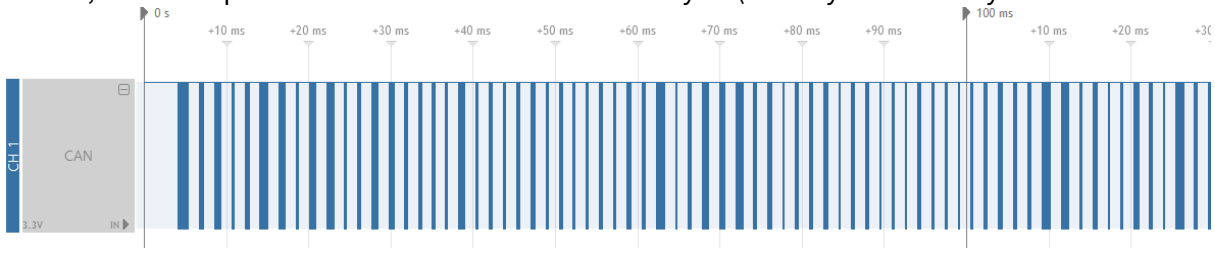


Figure 44: Example CAN bus signals displayed on the logic waveform

Zooming out on that same kind of signals shows the whole captures (as in the image below), but as explained before, make it impossible to read the value of decoded bytes (unless you constantly zoom in and out):



To make it easier to navigate through important amounts of decoded data packets and make sense out of its content, ScanaStudio offer special "Alternative" data views:

- Raw view
- Packet View
- Hex View

Those views can be accessed through ScanaStudio's main menu as shown in the image below:

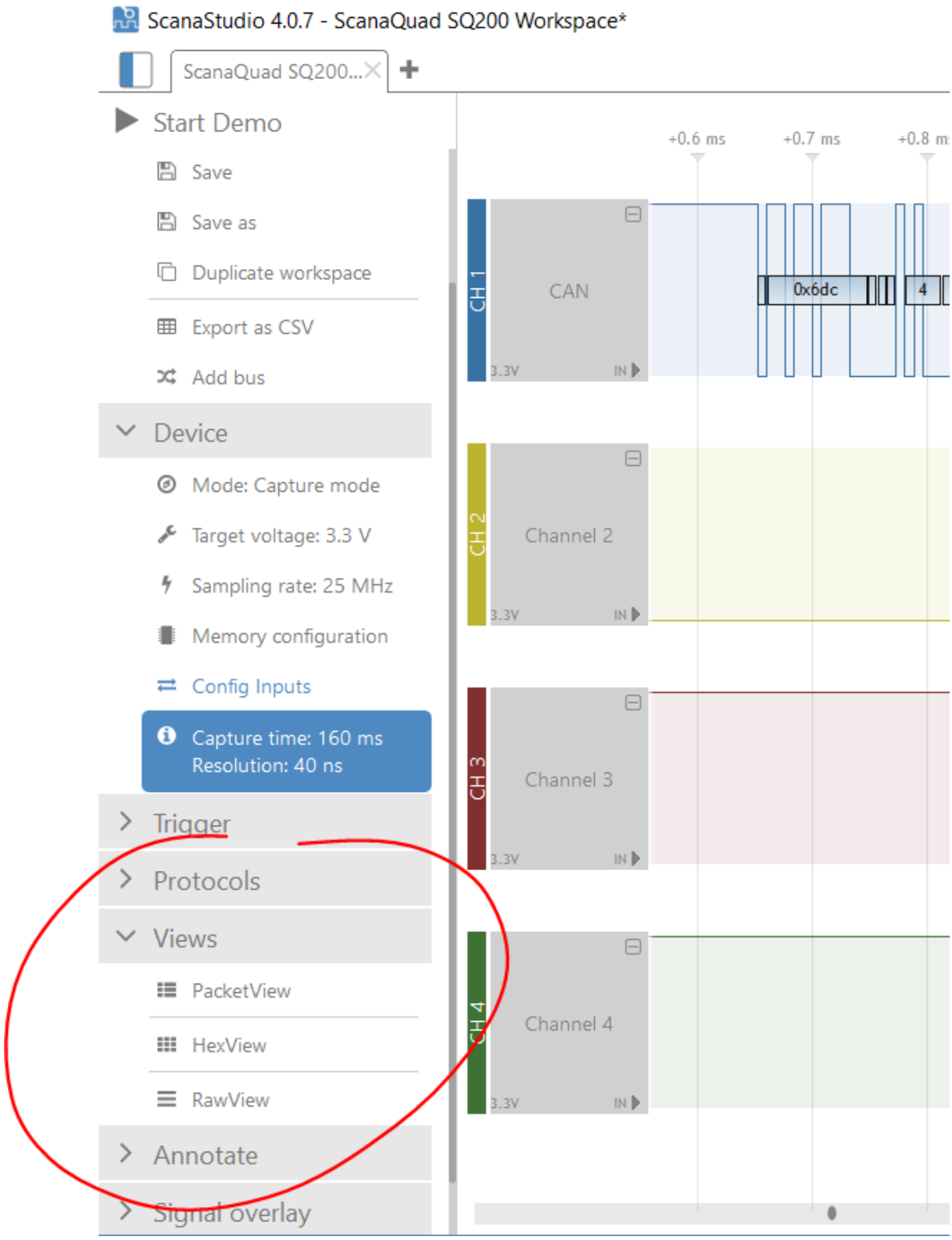


Figure 45: views menu in ScanaStudio

As it will be explained below, each and every one of those views have particular applications and advantages. As a matter of fact, not all protocol scripts implement all those views. It all depends on what makes sense for a specific protocol.

HEX View

The HEX view shows a memory-dump-like view, where each byte is given a row and column address. Also, ASCII equivalent characters are displayed when possible (otherwise, dots are displayed)

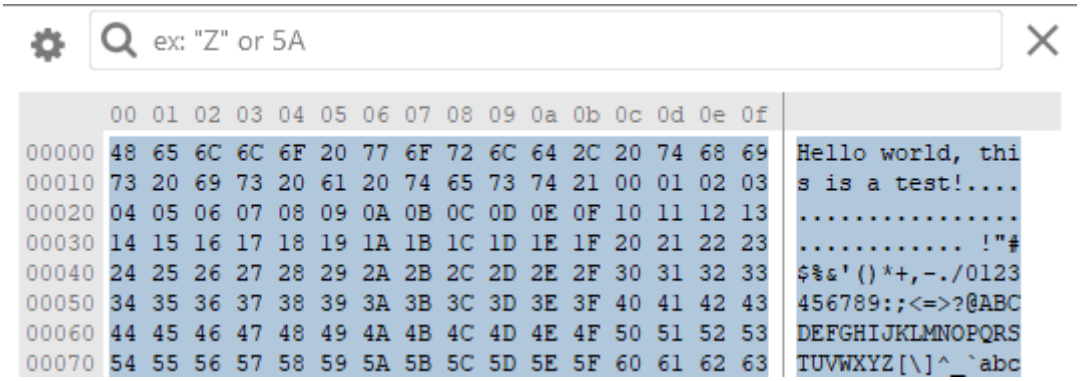


Figure 46: Hex view showing UART bytes

Obviously, HEX view is not practical for all applications and for all decoders. For example, If you have captured signals showing a firmware being updated via an UART interface, it can make a lot of sense to use the HEX view and adjust the number of columns in such a way to display firmware content in meaningful fashion (and compare it to intel formatted HEX files, for example). That being said, for other protocols like I2C, it may be more difficult to make good usage of the HEX view, and that's why other data views are provided.

Packet View

As the name implies, this view provides a way of visualizing captured and decoded data as packets:

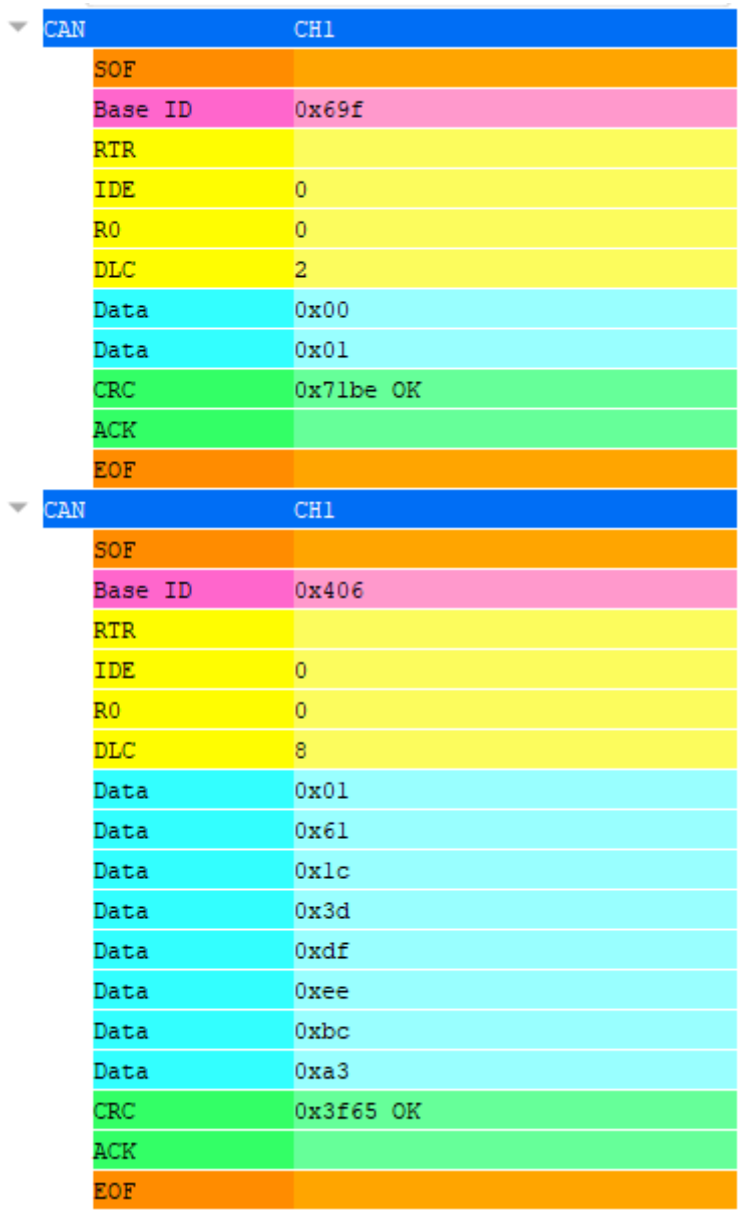


Figure 47: Example CAN bus packets

Just like the HEX View is not good for every protocol, Packet view is also adapted to specific protocols. For instance, UART protocol which does not define any structure for packets and instead just defines encoding of bytes wouldn't make much use of the Packet View, since there are no such thing as "packets" in UART. However when it comes to MODBUS protocol, which is a layer above UART protocol *and* defines

a structure for packets, it make sense to use the Packet View. Below is an example of MODBUS packets decoded and displayed in Paccket View.

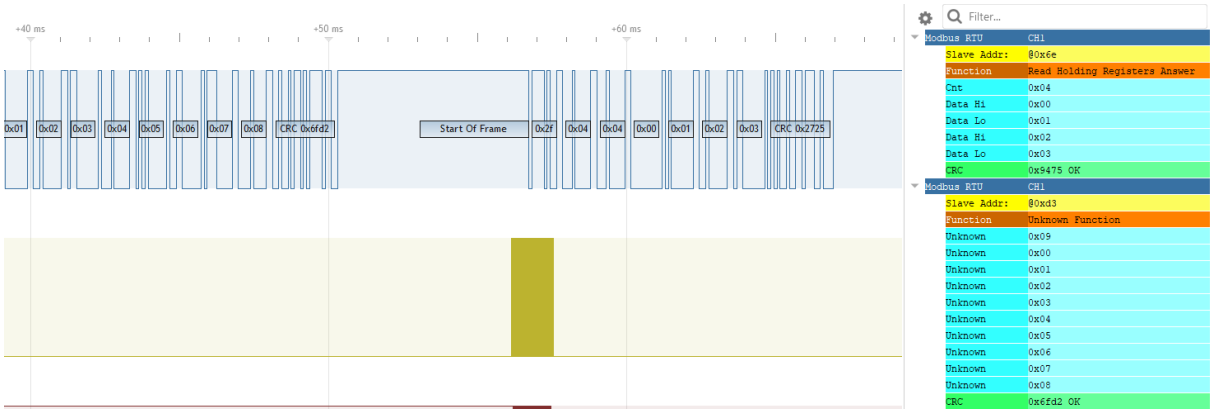


Figure 48: Example MODBUS packets

Double clicking on an item let's your pinpoint the actual logic signals related to that packet.

Raw View

The raw view is very simplistic and is implemented by-default for all protocol decoders. Raw view simply lists the decoded items, whether it's a start bit, a data byte or an acknowledge bit. Like other views, Raw Views can be filtered out to search for a specific item. Double clicking on an item let's your pinpoint the actual logic signals related to that raw view item.

https://www.ikalogic.com/pages/logic-analyzer-guide#!articles/1090-7768-exporting-data-and-signals

https://www.ikalogic.com/pages/logic-analyzer-guide#!articles/1090-7766-advanced-trigger-options

State mode external trigger industrial inputs

Generating signals: Composing and generating a simple square signal also talk about builder scripts

Using the mixed mode

What is timing incoherence and memory incoherence ?

https://www.ikalogic.com/pages/logic-analyzer-guide#!articles/1090-7772-signal-overlay

Scripts library manager