Czech Technical University in Prague, Faculty of Electrical Engineering
Department of Measurement, Laboratory of Videometry

VSVI - Versatile STM32 Virtual Instrument

User manual

Bc. Jozef Dujava

Contents

1 Introduction & getting started

The Versatile STM32 Virtual Instrument (VSVI) is a software-defined instrument platform implemented for multiple STM32 microcontrollers. The primary instrument is a mixed-signal oscilloscope with real-time and equivalent-time sampling capability. Additional instruments include pulse generators, arbitrary generators and a frequency counter. Very few additional components are necessary besides a supported STM32 microcontroller (MCU). The virtual instruments are operated from a PC using a terminal user interface (TUI) implemented within the MCU firmware. If supported, the MCU is connected to the PC via USB directly, otherwise an external USB-UART converter is used (e.g. CH340, ST-Link).

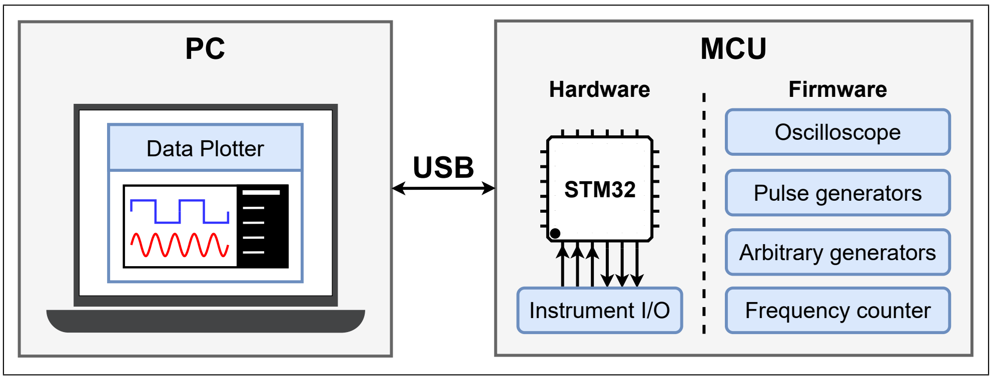


Figure 1: Diagram of the VSVI platform (USB version)

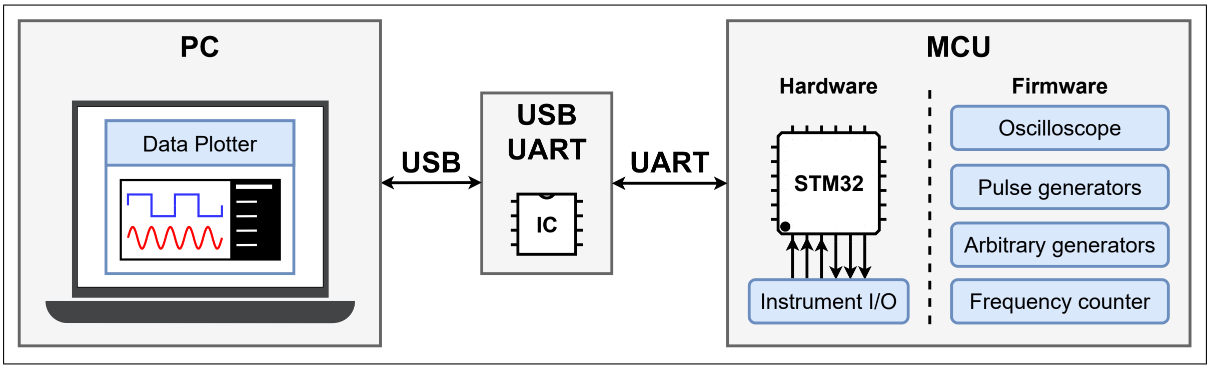


Figure 2: Diagram of the VSVI platform (UART version)

The universal Data Plotter PC application is used to display acquired oscilloscope waveforms, access the TUI and transfer files between the PC and microcontroller. This application was developed by Bc. Jiří Maier, and is not a part of my thesis. It is listed on the Embedded server[1] of the Department of Measurement, FEE, CTU in Prague. Its source code and binary releases are available from GitHub[2].

This manual describes the functionality and usage of the VSVI system. Note that certain features may only be available on higher-end MCUs, such as the STM32F303RE which is used as an example throughout this manual. Screenshots show the 17.3.2022 release of Data Plotter[3].

[1] https://embedded.fel.cvut.cz/platformy/dataplotter
[2] https://github.com/jirimaier/DataPlotter
[3] https://github.com/jirimaier/DataPlotter/releases/tag/v2.0

1.1 Programming the MCU

To use the VSVI platform, the microcontroller must be programmed with the appropriate firmware. This can be done via an ST-Link or serial (UART) programmer or directly via USB if a USB bootloader had been programmed previously onto the MCU. If UART is used to communicate with the PC, the baud rate is fixed within the firmware. Therefore, multiple firmware binaries had been compiled to support various baud rates. Binaries are named "**VSVI_<platform>_<comm>_<release>.bin**", where:

- **<platform>**: Name of the hardware platform. Can be either:
  - "<MCU code>−<package>", e.g. "G431-LQFP32" for standalone MCUs
  - "<MCU code>−<board name><pin count>", e.g. "G431-Nucleo32", "F303-Nucleo64", "F103-BluePill48" for development boards
- **<comm>**: Description of the PC communication interface, can be:
  - "UART−<baud>": using UART with baud rate <baud>
  - "USB−VCP": using USB in Virtual COM Port (VCP) mode.
- **<release>**: Release date in "R<YYYYMMDD>" format

1.2 Connecting the MCU to the PC

After the MCU has been programmed, connect it to the PC via the appropriate interface, according to the firmware version as described in the previous section. Also connect the necessary input and output signals according to the appropriate MCU pinout found in section 8. Open Data Plotter and select the MCU in the connection tab in the sidebar, as shown in Figure 3. The MCU's device descriptor should be listed in the drop-down menu, in this case it is "ttyACM0 - STM32 STLink".

If using UART, configure the correct baud rate in the connection window (921600 baud in this case). If using USB, the baud rate setting is irrelevant. Finally, click the connection button (indicated by cursor) to connect to the MCU. If the connection is successful, the MCU sends a welcome message which is displayed in the Data Plotter log. To start using the virtual instruments now, access the terminal user interface by switching to the terminal tab in Data Plotter.

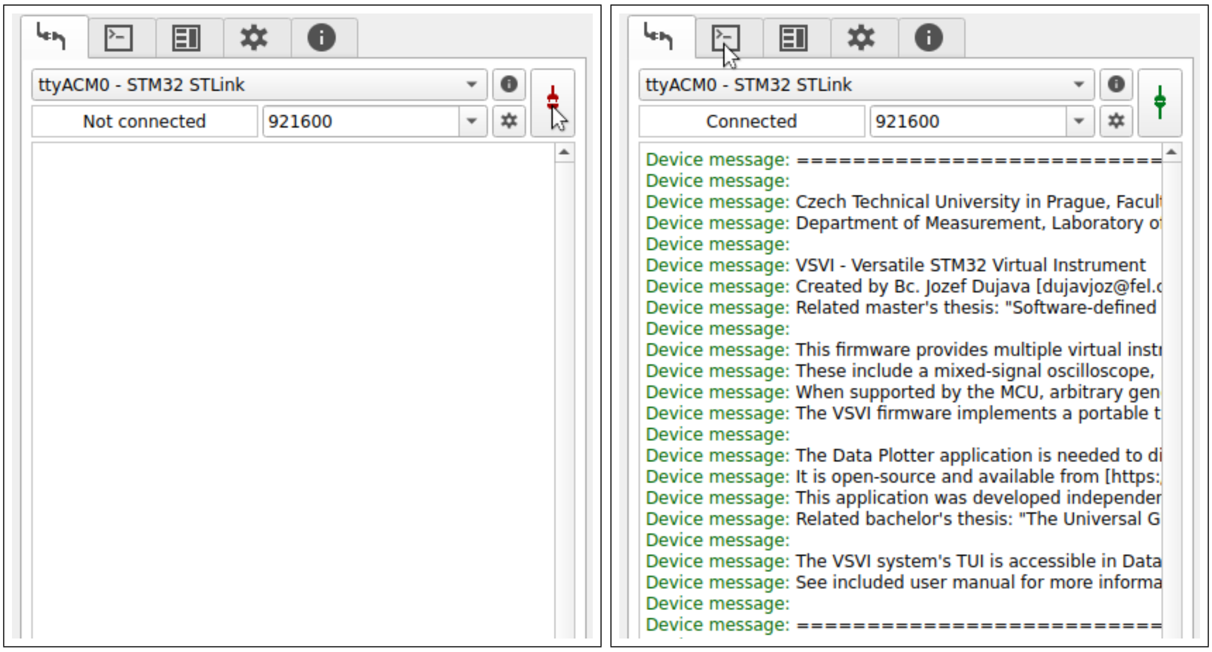


Figure 3: Data Plotter sidebar on startup (left) and after the MCU is connected (right)

2 Terminal user interface (TUI)

The terminal user interface (TUI) is displayed in Figure 4. It is compartmentalized into tabs, one for each type of virtual instrument. The name of the currently open tab is shown in the white title bar at the very top. The blue button to the left of the title opens a drop-down tab selection menu containing the name of every tab. Clicking the corresponding button on the left then opens this tab.

Within each instrument tab, there are a number of settings/parameters displayed. Each group of settings is described in detail in the ”¡instrument code¿ TUI tab:” sections of this manual. Values are highlighted in white blocks, with units immediately on their right. If the value of a parameter can be changed by the user, a typically blue ”edit” button is also shown to the left of the parameter's name. Its behavior depends on the parameter type, detailed in the following subsection.

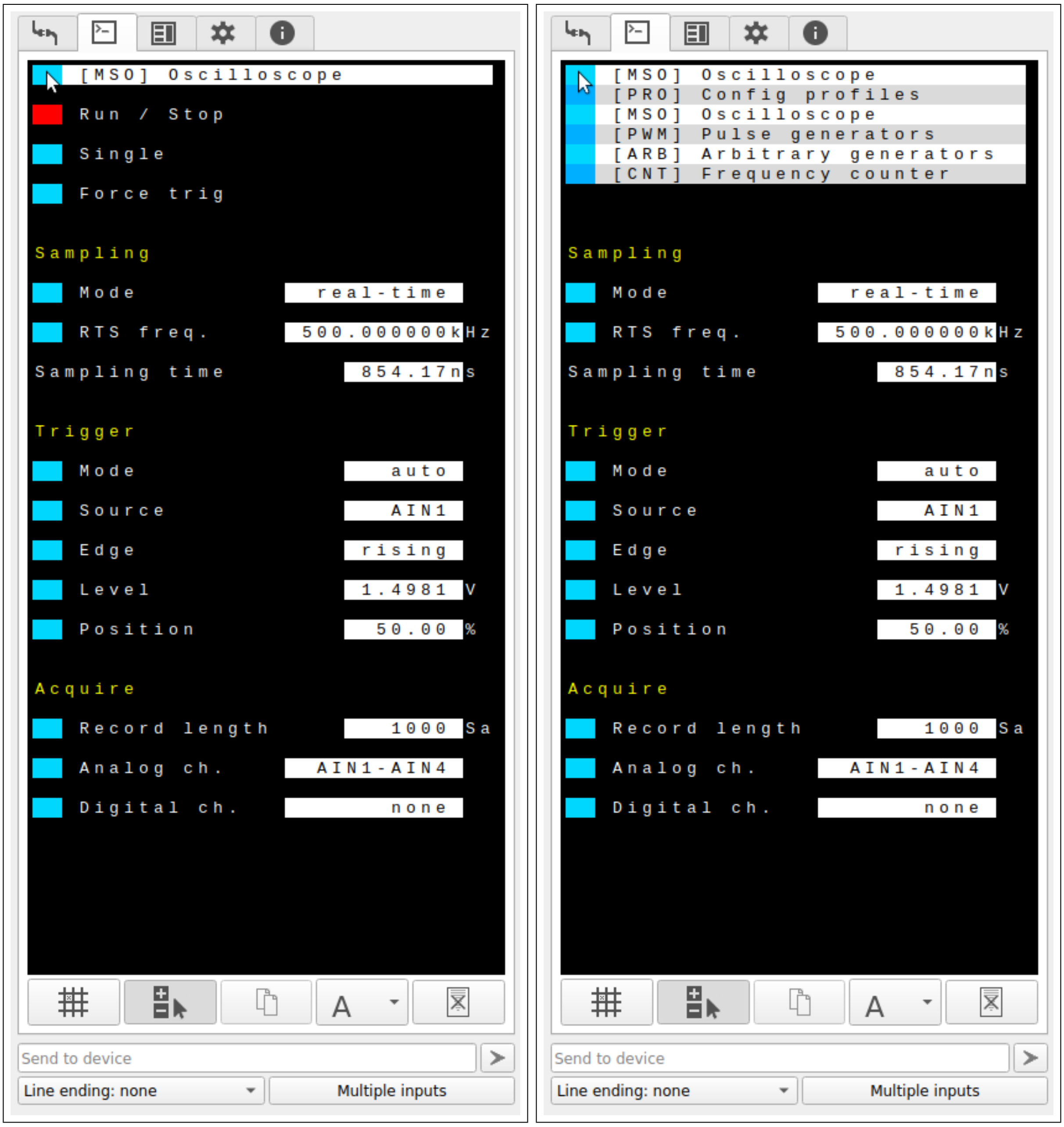


Figure 4: Oscilloscope tab of the TUI (left), with tab menu opened (right)

Metric unit prefixes are supported for numeric values, consisting of a single character at the very end of the value, e.g. ”12.345k”. The available prefixes are listed in the following table. Characters other than digits, metric prefix characters and the decimal point are ignored.

Prefix	nano	micro	mili	-	kilo	mega	giga
Character	”n”	”u”	”m”	” ”	”k”	”M”	”G”
Multiplier	10^{-9}	10^{-6}	10^{-3}	1	10^{3}	10^{6}	10^{9}

Table 1: Supported metric unit prefixes

In many cases, the exact parameter value entered by the user cannot be achieved. The firmware then sets the closest possible value, which is shown in the TUI. However, the originally entered value is saved and the firmware will attempt to achieve a closer match if the instrument configuration is changed later.

2.1 Value editing

Parameter values can be edited in multiple ways, depending on their type. The supported types are:

- **Toggle settings**: For settings with only a few discrete options, e.g. ”on”/”off”, clicking the edit button repeatedly cycles through the available options. The new value is applied immediately, there is no confirmation.

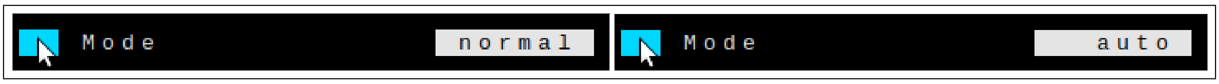


Figure 5: Example of a toggle setting (”Mode” setting of the oscilloscope trigger)

- **Drop-down menu**: For settings with more discrete options, a drop-down menu is opened underneath the parameter's value block when the edit button is clicked. This menu lists all the available options, each with a button on the left side. Clicking one of these buttons selects the corresponding option, applies it immediately and closes the menu. The menu is also closed when the user clicks any parameter edit button, keeping the old value.

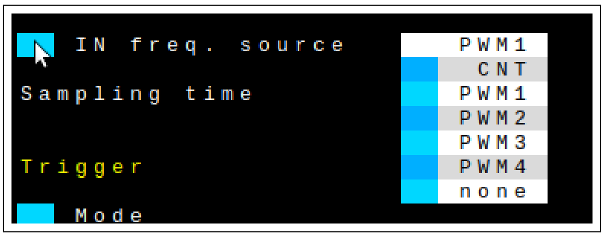


Figure 6: Example of the drop-down menu (”IN freq. source” setting of the oscilloscope)

- **Numeric editor**: A digit-wise editor displayed in a row under the value block is provided for numeric values. The ”<” and ”>” buttons select a digit in the value. The ”+” and ”-” buttons then increment/decrement the selected digit or cycle through available metric prefixes if the rightmost digit position is selected. The new value is saved when the user clicks any parameter edit button. There may also be a number of presets available below, which work identically to the drop-down menu described above.

Figure 7: Example of numeric editor with presets ("RTS freq." setting of the oscilloscope)

- **Direct value entry (text field):** Lastly, it is also possible to enter values directly using the "Send to device" function of Data Plotter located below the terminal window. This text field is pre-filled with the current parameter value when the value editor (drop-down or numeric) is opened.

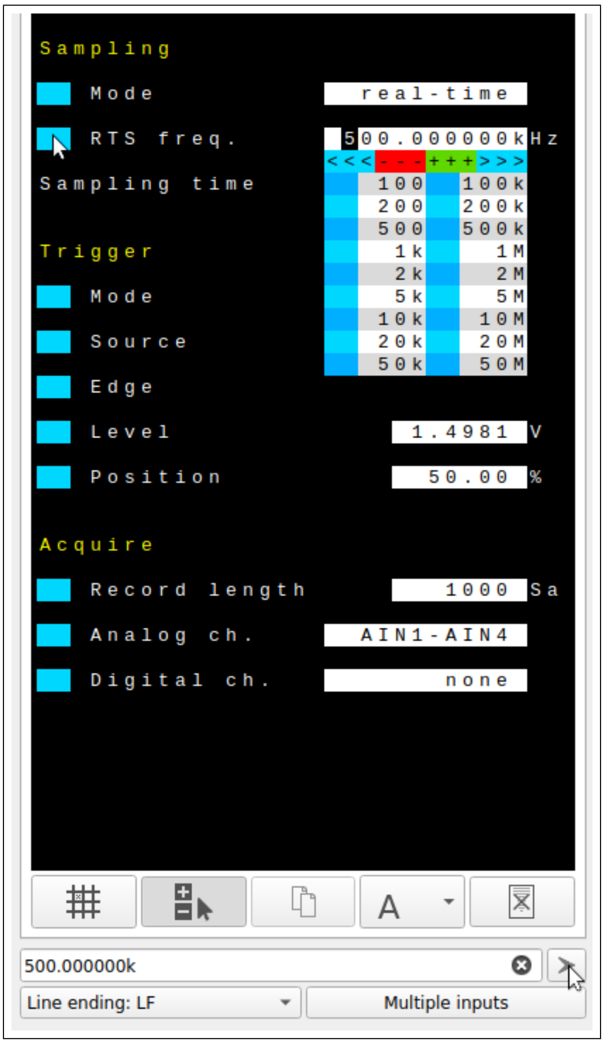


Figure 8: Pre-filled text field of the "Send to device" function for direct value entry

3 Mixed-signal oscilloscope (MSO)

A Mixed-Signal Oscilloscope (MSO) with 4 analog channels and up to 16 digital channels is implemented. Its exact capabilities depend on the MCU used. The Analog-to-Digital Converters (ADCs) embedded in the MCU are used for the analog channels. General-Purpose Input/Output (GPIO) pins in input mode are used for the digital channels. Both the analog and digital channel sampling is triggered by an embedded timer at a precise sampling frequency. Samples are transferred by DMA controllers into buffers in internal SRAM memory. After an acquisition is completed, the buffer contents are sent to the PC in order to display the acquired data in Data Plotter. Both real-time and equivalent-time sampling are implemented.

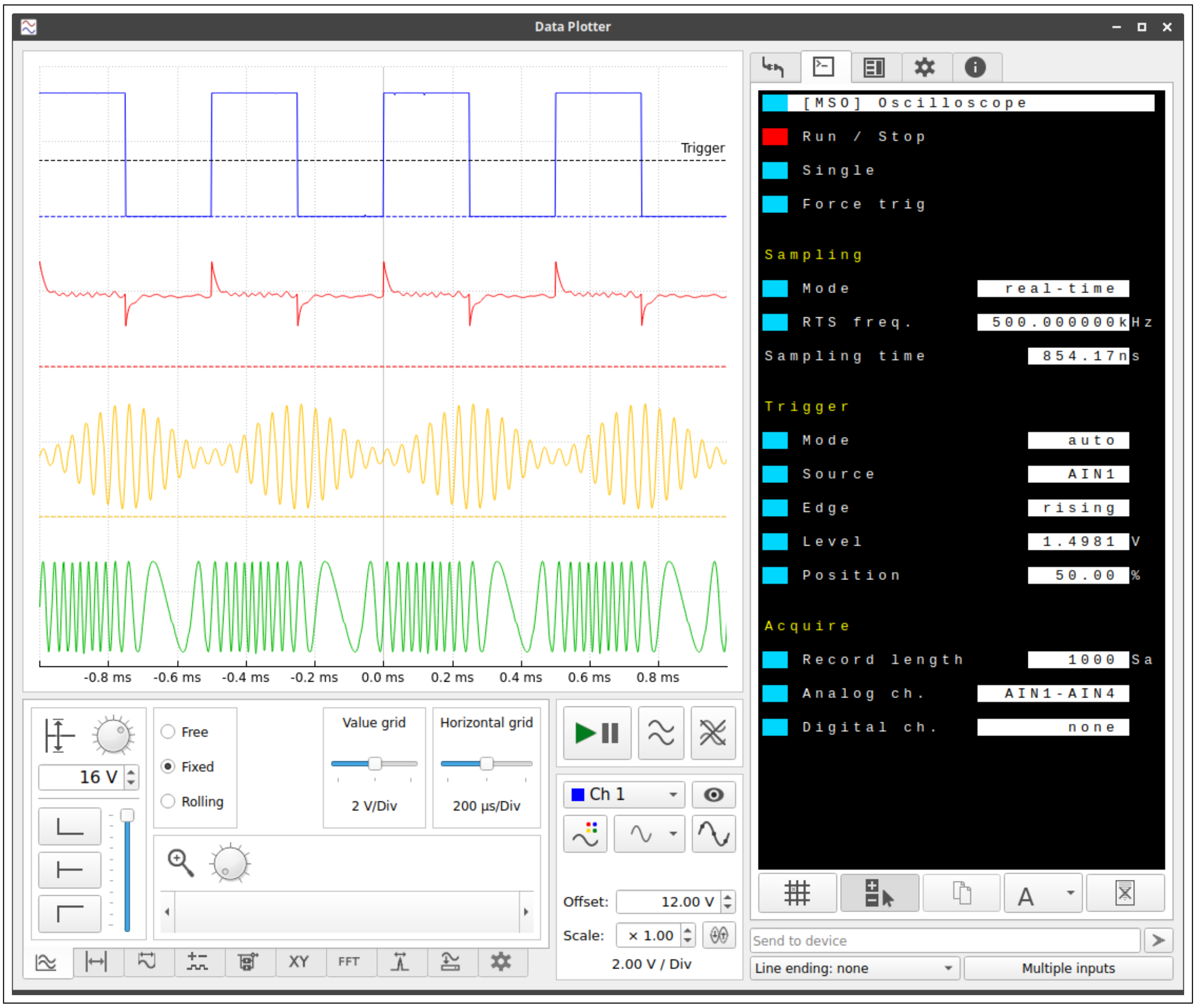


Figure 9: Oscilloscope being used within Data Plotter

Note that the acquisition is paused whenever the user is interacting with the TUI. This includes the time it takes to process a button click, as well as the entire time the value editor is open. This applies even when interacting with a different instrument's tab. This restriction is necessary to ensure communication with the PC does not disturb the operation of the oscilloscope at high sampling rates. Once the setting is stored and editor closed, the acquisition is automatically restarted.

In RTS mode, the ADCs sample the inputs at a sampling frequency f_{SR} set by the "RTS freq." parameter. The maximum sampling rate in this mode is not only limited by the maximum ADC sampling rate, but also other factors such as DMA throughput. For example, the maximum sampling rate is lower when an arbitrary generator is enabled. If the user enables more analog channels (N_C) than there are ADCs available (N_A):

- Each ADC samples multiple input channels alternately.
- The maximum sampling frequency is divided by a factor of

$$N_{alt} = \frac{N_C}{N_A}. \tag{1}$$

- Channel skew (delay) of no more than

$$T_{skew} = \frac{T_{SR}}{N_{alt}} = \frac{1}{N_{alt} \cdot f_{SR}} \tag{2}$$

  is introduced between channels sampled by the same ADC.

In summary:

- For MCUs with 4 ADCs ($N_A = 4$):
  - There is no sampling rate reduction nor channel skew.
- For MCUs with 2 ADCs ($N_A = 2$):
  - For 1 or 2 enabled analog channels ($N_C \leq 2,\ N_{alt} \leq 1$):
    * There is no sampling rate reduction nor channel skew.
  - For 4 enabled analog channels ($N_C = 4,\ N_{alt} = 2$):
    * Max. sampling rate is reduced to 1/2 of the max. sampling rate for $N_C \leq 2$.
    * Channels 1 and 2 are sampled simultaneously.
    * Channels 3 and 4 are sampled simultaneously.
    * Channels 3 and 4 are are delayed from channels 1 and 2 by at most 1/2 of the sampling period.
- For MCUs with 1 ADC ($N_A = 1$):
  - For 1 enabled analog channel ($N_C = 1,\ N_{alt} = 1$):
    * There is no sampling rate reduction nor channel skew.
  - For 2 enabled analog channels ($N_C = 2,\ N_{alt} = 2$):
    * Max. sampling rate is reduced to 1/2 of the max. sampling rate for $N_C = 1$.
    * Channel 2 is delayed from channel 1 by at most 1/2 of the sampling period.
  - For 4 enabled analog channels ($N_C = 4,\ N_{alt} = 4$):
    * Max. sampling rate is reduced to 1/4 of the max. sampling rate for $N_C = 1$.
    * Each channel is delayed from the previous one by at most 1/4 of the sampling period.

3.1 Interleaving ADCs

If there are more available ADCs than enabled analog channels ($N_A > N_C, N_{alt} < 1$), ADC interleaving can be used to achieve a higher sampling rate. This is activated automatically when the user selects a real-time sampling frequency that is beyond the capabilities of a single ADC. The "Interleaving" parameter appears, displaying the number of ADCs interleaved per analog channel, e.g. "2 ADCs/ch". If the sampling frequency is subsequently decreased, interleaving is disabled again.

In some MCUs, the firmware cannot connect the input pins of analog channels to all the interleaving ADCs. Therefore, external jumper links between the analog input pins are required. In that case, a "Links needed" parameter appears, listing the required connections. These are shown with a yellow background to alert the user to the necessity of connecting the jumper wires. If the user fails to do so, samples of different input signals will be combined into one channel, producing garbled waveform data.

3.1.1 Trigger limitations

When normal trigger mode from an analog channel is used, only one ADC can use the analog watchdog triggering feature due to limited AHB bandwidth and core speed of the MCU. Therefore, while the effective sampling rate f_S is higher due to using

$$N_{int} = \frac{1}{N_{alt}} \tag{3}$$

interleaving ADCs per analog channel, the rate at which the trigger can detect signal transitions is still equal to the sampling rate f_{S1} of a single ADC, i.e.

$$f_{S1} = \frac{f_S}{N_{int}}. \tag{4}$$

Therefore, trigger aliasing (using 1 ADC only) will occur before waveform aliasing (using multiple ADCs). Aliasing occurs when the sampling frequency is at or below the Nyquist frequency. When f_{S1} is exactly the Nyquist frequency, all the samples from the trigger ADC will be constant and no trigger will occur at all. When f_{S1} is below the Nyquist frequency, trigger events will happen at a reduced rate, missing some valid signal transitions.

3.2 Digital channel skew

The digital channels may be slightly delayed with respect to analog channels. The digital data is captured simply by a DMA transfer from the GPIO input data register, there is no sample and hold mechanism. Ongoing transfers of ADC data for analog channels or DAC data for arbitrary generators occupy the AHB bus and cause delays of variable durations. Unfortunately, there is no way to alleviate this without additional input latch circuitry.

3.3 Equivalent-time sampling (ETS) mode

Equivalent-time sampling (ETS) mode is supported for periodic input signals. Their frequency f_{IN} must be known for this implementation. The input frequency can be measured by the frequency counter or manually entered. The input signal can also be synchronous with an output signal of one of the available generators, the frequency of which is known. Then, an equivalent-time sampling rate f_{SE} higher than the maximum f_{SR} can be achieved by acquiring one sample per N input signal periods according to

$$T_{SR} = N \cdot T_{IN} + T_{SE} \iff f_{SR} = \frac{f_{IN} \cdot f_{SE}}{f_{IN} + N \cdot f_{SE}} \tag{5}$$

$$T_{SE} = T_{SR} - N \cdot T_{IN} \iff f_{SE} = \frac{f_{IN} \cdot f_{SR}}{f_{IN} - N \cdot f_{SR}} \tag{6}$$

where T_{IN}, T_{SR}, T_{SE} are the input, RTS and ETS periods respectively.

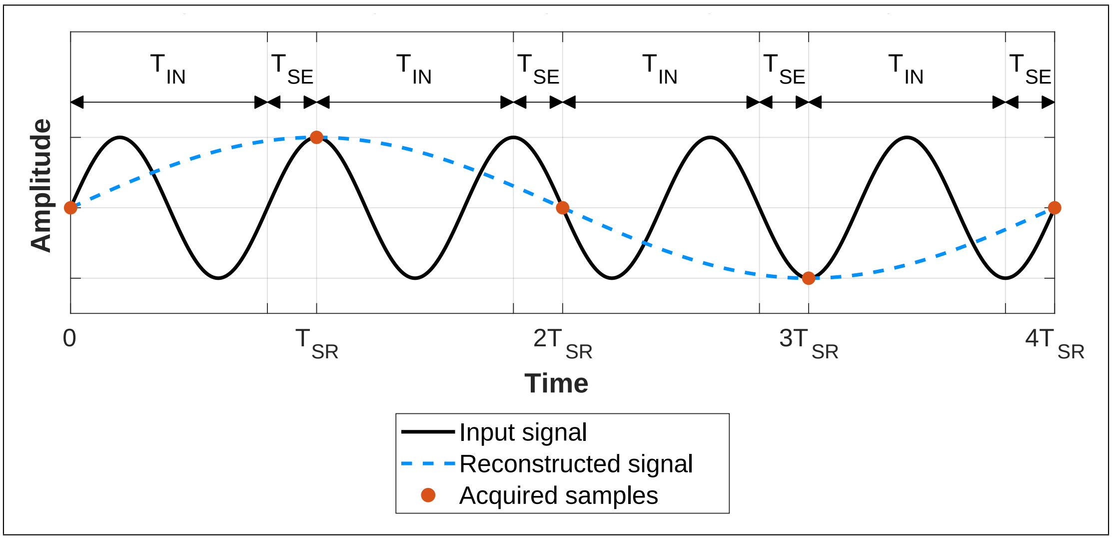


Figure 10: Equivalent-time sampling (ETS) as implemented in this firmware. Sampling at an equivalent of 4 samples per period with $N = 1$.

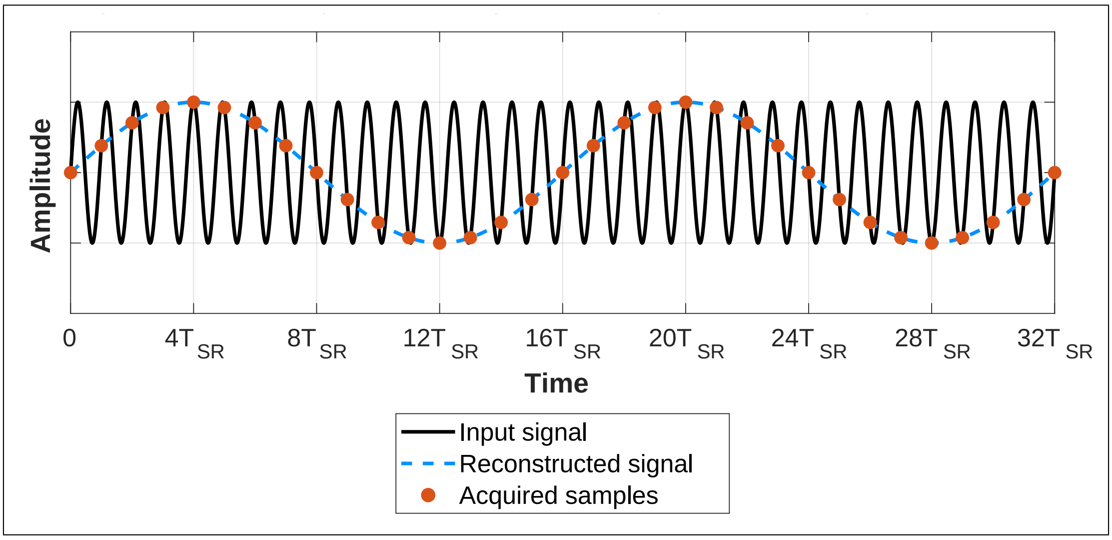


Figure 11: Equivalent-time sampling (ETS) at an equivalent of 16 samples per period with $N = 1$.

3.3.1 Trigger limitations (digital channels)

Input signal edge detection of the EXTI controller is used for digital channel trigger. In ETS mode, this will cause detection events to happen between samples, stopping oscilloscope acquistion prematurely. In normal trigger mode, if the selected edge cannot be found in the acquired data, the oscilloscope discards it and starts a new acquistion without updating the waveform displayed. This would cause the oscilloscope to appear frozen if a digital channel is used as trigger source in ETS mode with normal trigger. Therefore, there are two options for triggering in ETS mode:

- Enable at least one analog channel and use it as the trigger source. Both normal and auto trigger are available and work normally.
- If a digital channel is used, only auto trigger is available. However, input signal transitions are not detected, trigger only occurs after a number of full buffers have been acquired. For this reason, the trigger edge selection is hidden for digital channel trigger in ETS mode.

3.4 MSO TUI tab: Main controls

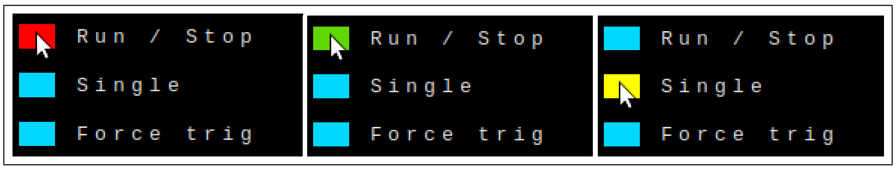


Figure 12: Main oscilloscope controls in Stop, Run and Single mode (from left to right)

3.4.1 Run/Stop

Toggling this setting starts and stops the oscilloscope acquisitions. The button is red in Stop mode, green in Run mode and blue in Single mode.

3.4.2 Single

Clicking this button enables single mode for the next acquisition, reverting to Stop mode afterwards. The button is yellow when single mode is active and waiting for trigger, otherwise blue.

3.4.3 Force trig

Forces a single trigger event to occur when clicked.

3.5 MSO TUI tab: "Sampling" menu

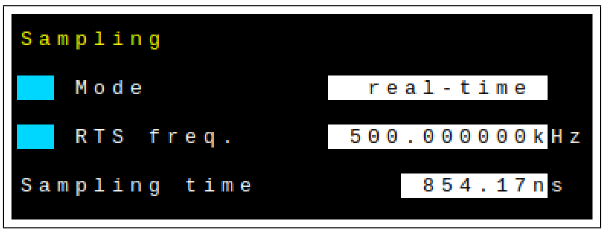


Figure 13: Sampling menu in RTS mode without ADC interleaving

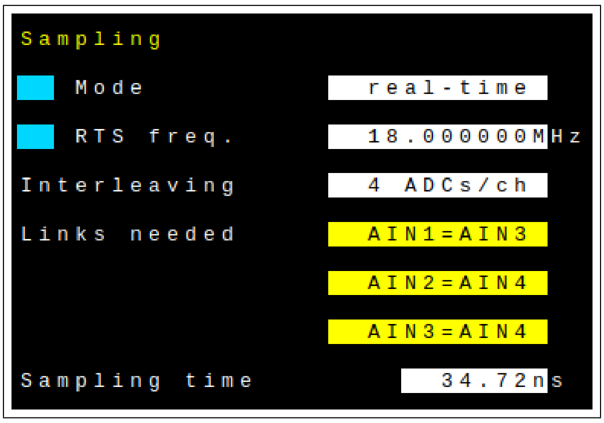


Figure 14: Sampling menu in RTS mode with active ADC interleaving

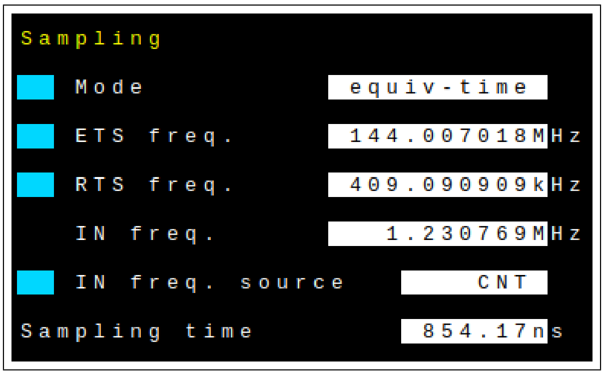


Figure 15: Sampling menu in ETS mode

3.5.1 Mode

- "real−time": Oscilloscope uses real-time sampling (RTS mode).
- "equiv−time": Oscilloscope uses equivalent-time sampling (ETS mode).

3.5.2 RTS freq.

Real-time sampling frequency, in Hertz. This is the effective sampling rate including interleaving of ADCs (each individual ADC samples at the corresponding fraction of this frequency when interleaving is active).

3.5.3 Interleaving

Currently active interleaving mode. Only shown when interleaving is active. Selected automatically based on the sampling rate and number of enabled analog channels, not user-configurable. The options are:

- "2 ADCs/ch": There are 2 ADCs sampling every enabled analog channel.
- "4 ADCs/ch": There are 4 ADCs sampling every enabled analog channel.

3.5.4 Links needed

A list of jumper links necessary for the currently active interleaving mode. A link connecting analog input pins "AIN X" and "AIN Y" is denoted as "AINX=AINY". Only shown when interleaving is active and the analog inputs cannot be connected by firmware.

3.5.5 ETS freq.

Equivalent-time sampling frequency, in Hertz. Only applicable and visible in ETS mode. This is the equivalent sampling rate for an input signal with frequency equal to the value of the "IN freq." parameter. There are 2 ways this parameter can work:

- The user can set the "ETS freq." parameter directly. The firmware then finds an RTS frequency such that the resulting ETS frequency is as close as possible to the entered value. If the input frequency changes, the firmware readjusts the RTS frequency to maintain the desired ETS frequency.
- Alternatively, the user can set the "RTS freq." parameter. In that case, the "ETS freq." parameter only shows the calculated ETS frequency, based on the input frequency. If the input frequency changes, the RTS frequency remains unchanged, while the ETS frequency is recalculated.

3.5.6 IN freq.

Input signal frequency, in Hertz. Only applicable and visible in ETS mode. This parameter can be set manually when the input frequency source ("IN freq. source" parameter) is set to "none". Otherwise, it updates automatically from the selected source and can't be changed by the user.

3.5.7 IN freq. source

Source of input signal frequency with the following options:

- "CNT": Input signal frequency measured by the frequency counter. Automatically updated at the counter's update rate (approx. 1 second).
- "PWMX": Input signal frequency is set equal to the pulse generator "X" frequency. Automatically updated whenever the user changes the generator's frequency.
- "ARBX": Input signal frequency is set equal to the arbitrary generator "X" frequency. Automatically updated whenever the user changes the generator's frequency.
- "none": No automatically updated source. Input signal frequency can be set manually in the "IN freq." parameter.

3.5.8 Sampling time

The ADC sampling time, in seconds. This is the duration of the sampling phase of the ADC, during which the sampling capacitor charges from the input pin. It determines the maximum input signal impedance R_{AIN} for a given ADC resolution. The corresponding values can be found in the MCU datasheet. **The sampling time is automatically set as high as possible for a given sampling rate**, it is not user-configurable.

3.6 MSO TUI tab: "Trigger" menu

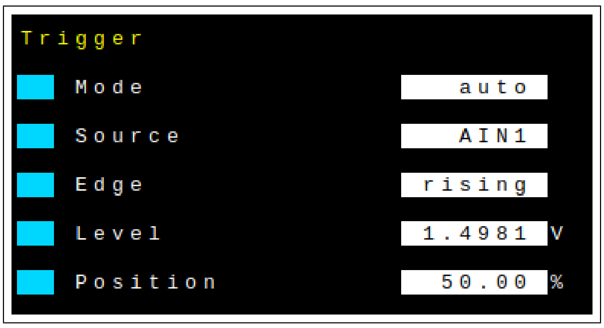


Figure 16: Trigger menu, set to trigger from an analog channel

3.6.1 Mode

- "normal": Trigger when a transition of the input signal occurs.
- "auto": Trigger on a transition, or when 5 full buffers have been acquired.

3.6.2 Source

- "AINX": Analog channel X (input pin AIN X) triggers the oscilloscope.
- "DINX": Digital channel X (input pin DIN X) triggers the oscilloscope.

Only enabled channels are available in the drop-down menu.

3.6.3 Edge

- "rising": Trigger on a rising edge only.
- "falling": Trigger on a falling edge only.
- "either": Trigger on either a rising or falling edge.

3.6.4 Level

Vertical trigger level, in Volts. Hidden when triggering from a digital channel.

3.6.5 Position

Horizontal position of the trigger (pre-trigger) as a percentage of the record length.

3.7 MSO TUI tab: "Acquire" menu

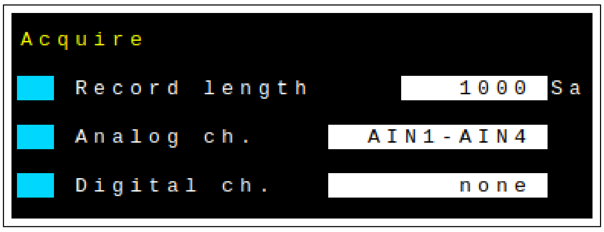


Figure 17: Acquire menu

3.7.1 Record length

Record length, as the number of samples acquired.

3.7.2 Analog ch.

Selection of enabled analog channels, the options are:

- "none": All analog channels are disabled. The digital channels must be enabled in this case.
- "AIN1": Only analog channel 1 is enabled.
- "AIN1,AIN2": Analog channels 1 and 2 are enabled.
- "AIN1−AIN4": All analog channels (1 to 4) are enabled.

3.7.3 Digital ch.

Selection of enabled digital channels, the options are:

- "none": All digital channels are disabled. At least one analog channel must be enabled in this case.
- "DIN0−DINX": All digital channels (0 to X) are enabled.

4 Pulse generators (PWM)

One or more pulse generators (depending on MCU capabilities) are also provided by this firmware. These can be used for general-purpose tasks or in conjunction with the oscilloscope in ETS mode, generating reference signals with precisely known frequencies. All generated frequencies are fractions of the MCU timer clock frequency f_{TIM} (typically equal to the system clock frequency f_{SYS}). Therefore, the available frequency resolution is diminished when approaching the timer clock frequency. Theoretically, the maximum output frequency is half the timer clock frequency. However, it may not be possible to drive the corresponding GPIO output pin at that frequency, depending on the load capacitance and other factors – consult the MCU datasheet for more details.

The pulse generators are denoted as ”PWM” and numbered (”PWM 1”, ”PWM 2”, etc.). Each generator can have one or more output channels, denoted by a letter suffix (”PWM 1A”, ”PWM 1B”, ”PWM 2A”, etc.). The output signal's positive duty cycle/pulse width can be set individually for each channel, while the frequency is the same for all channels of a given generator. Each channel's output can be individually disabled, leaving the corresponding GPIO pin floating (high-Z state).

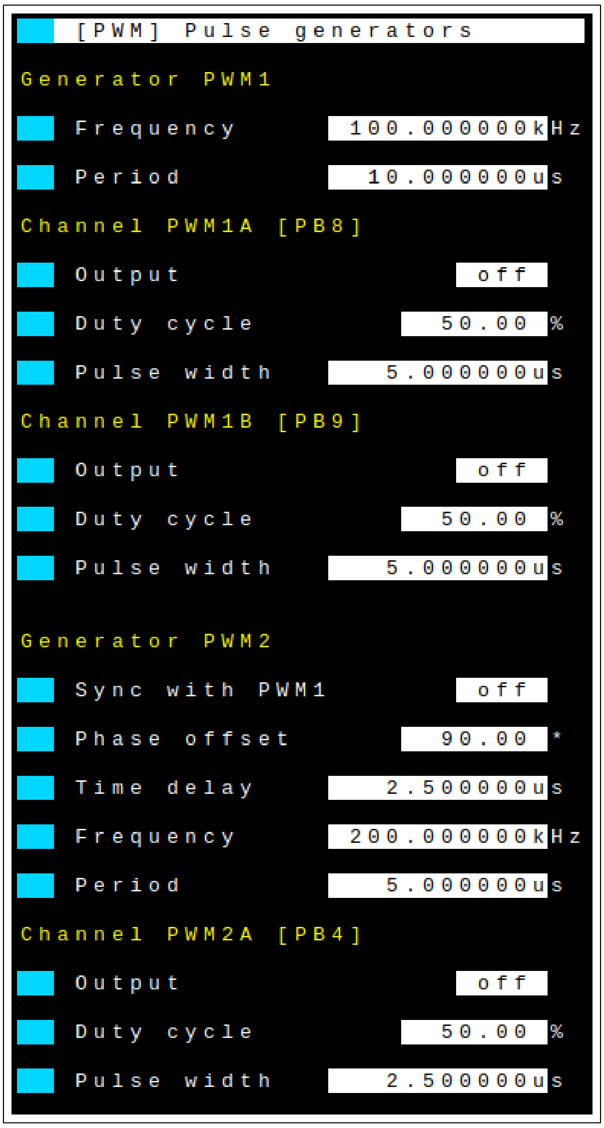


Figure 18: Pulse generator TUI tab

4.1 PWM synchronization

If there are multiple pulse generators available, it is possible to synchronize them in a master/slave scheme. The time delay of the slave generator's output signals w.r.t. the master generator's is configurable from zero (synchronized) up to the period of the master generator. A phase offset based on the period of the master generator can also be set. Additionally, frequency synchronization can be enabled via the "Sync with PWM X" setting. The frequency of the slave generator is then automatically kept equal to the master generator's frequency.

Each generator can be a slave of the previous generator, i.e. the PWM 3 generator can be a slave of PWM 2, which itself can be a slave of PWM 1. **Any number of generators can be synchronized this way, there are no additional unwanted delays introduced.**

4.2 PWM TUI tab: "Generator PWMX" menus

These menus contain all parameters specific to a given PWM X pulse generator.

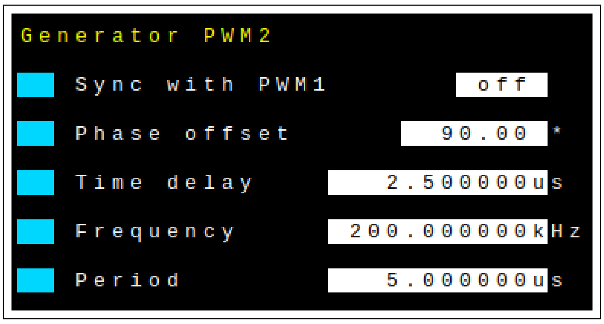


Figure 19: Generator parameters when frequency synchronization is disabled

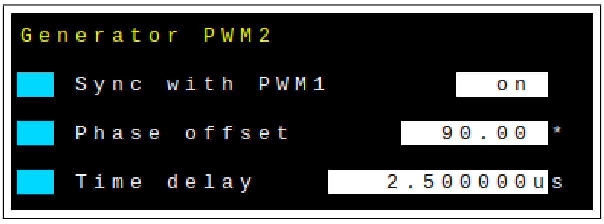


Figure 20: Generator parameters when frequency synchronization is enabled

4.2.1 Sync with PWMX

- "off": Frequency synchronization is disabled. This generator's frequency can be set independently.
- "on": Frequency synchronization is enabled. This generator's frequency is kept equal to the frequency of its master (PWM X).

4.2.2 Frequency

Generator frequency, in Hertz. Only visible when frequency synchronization is disabled.

4.2.3 Period

Generator period, in seconds. Only visible when frequency synchronization is disabled.

4.2.4 Phase offset

Phase offset of the slave generator w.r.t. the master generator, in degrees. Note the phase angle is relative to the period of the *master* generator – i.e. a 360° offset represents a delay of 1 master period.

4.2.5 Time delay

Time delay of the slave generator w.r.t. the master generator, in seconds. Maximum delay is 1 master generator period.

4.3 PWM TUI tab: "Channel PWMXY" menus

These menus contain all parameters specific to a given PWM XY output channel.

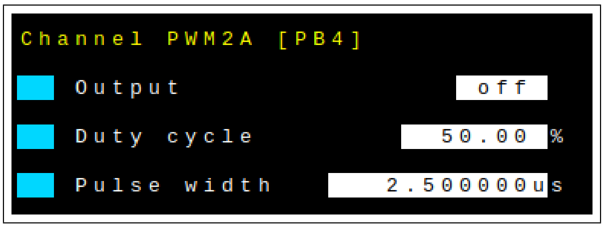


Figure 21: Channel parameters

4.3.1 Output

- "off": Output is disabled, leaving this channel's GPIO output pin floating (high-Z state).
- "on": Output is enabled, driving this channel's GPIO output pin.

4.3.2 Duty cycle

Positive duty cycle of the channel's output signal, as a percentage of the generator period.

4.3.3 Pulse width

Positive pulse width of the channel's output signal, in seconds.

5 Arbitrary generators (ARB)

One or two arbitrary generators are available in firmware versions for MCUs with embedded Digital-to-Analog Converters (DACs). These allow the generation of arbitrary signals as configured by the user. They are denoted as "ARB" and numbered (e.g. "ARB 1", "ARB 2"). The DAC sampling frequency f_S is generated by a timer, same as the pulse generators. The output signal frequency f_O is then

$$f_O = \frac{f_S}{N}, \tag{7}$$

where N is the number of samples per period of generated signal. The minimum number of samples per period is set to **4**, giving a maximum generated signal frequency of **250 kHz** when considering the typical max. DAC sampling rate of **1 MSa/s**. The max. number of samples per period (at low output signal frequencies) is typically **1000**, but it may depend on the firmware version. Between these extremes, the number of samples is automatically set to the highest value possible (not user-configurable).

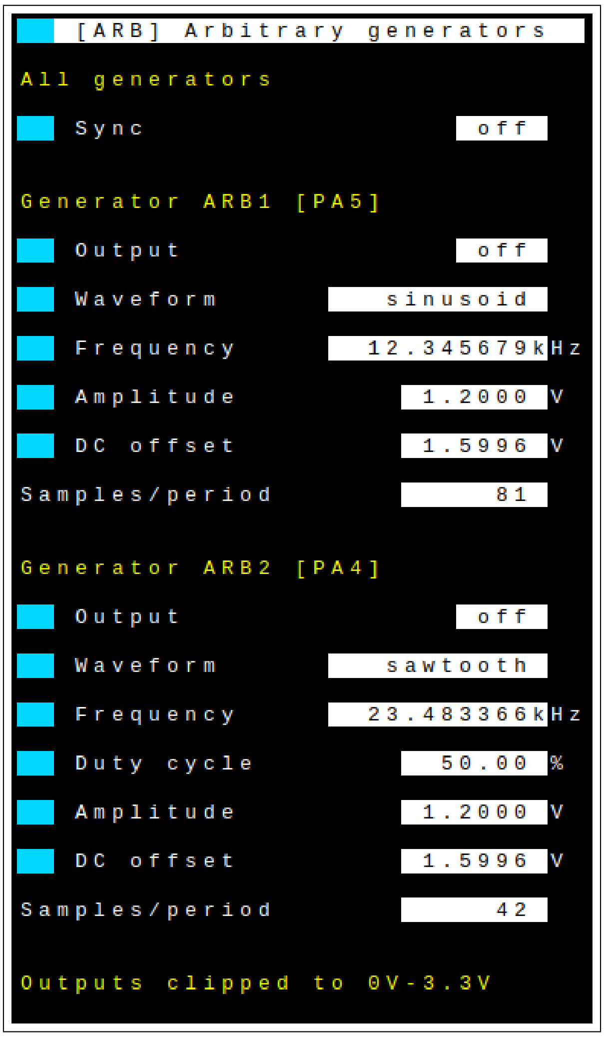


Figure 22: Arbitrary generator TUI tab

5.1 Custom waveforms

The arbitrary generators provide a number of built-in waveform functions (sine wave, sawtooth, etc.), but custom waveforms can also be used. These have to be loaded from the PC using a wave file in accordance with the following format:

- A comma-separated list of numeric values (.csv).
- Each value represents a single sample of the output signal, in Volts.
- The length of a single value (excl. comma) cannot exceed 62 characters.
- Metric prefix characters (e.g. '123m' for 123 mV) are supported.
- Characters other than commas, digits, metric prefix characters and the decimal point are ignored.
- Values will be clipped to remain within the generator's output voltage range.
- If there are more values than can be used by the generator, they will be ignored.

After a waveform has been loaded into the MCU, it can be retained between MCU resets by saving a configuration profile to the internal FLASH memory – see section 7.

5.2 ARB Synchronization

If there are two available arbitrary generators, they can be synchronized using the "Sync" setting. When enabled, both generators share a common frequency with an identical number of samples per period. The initial phase of both generators can then be set. Its resolution is determined by the number of samples per period, giving a worse resolution at higher frequencies (same as duty cycle).

If one of the generators is using a custom waveform, the other generator will be limited to the same number of samples per period. If both are using custom waveforms, the lower number of samples among them will be used for both – truncating the longer waveform.

5.3 ARB TUI tab: "All generators" menu

This menu contains settings affecting all arbitrary generators.

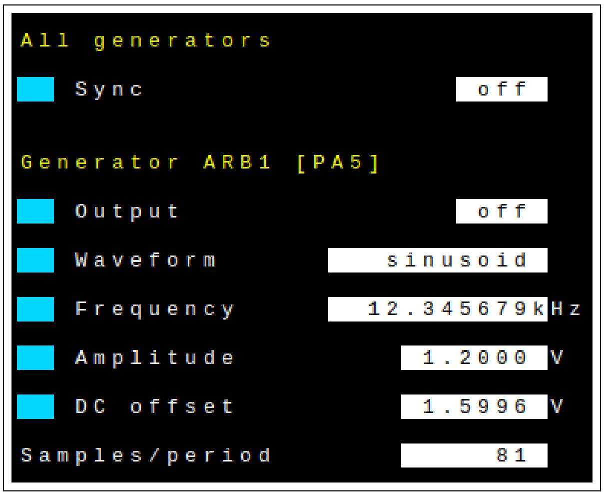


Figure 23: Generator parameters when synchronization is disabled

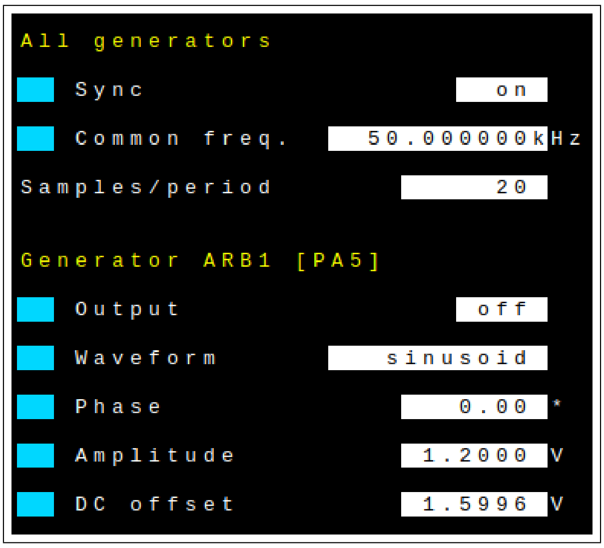


Figure 24: Generator parameters when synchronization is enabled

5.3.1 Sync

- ”on”: Synchronization is enabled, ARB 1 and ARB 2 frequencies are identical.
- ”off”: Synchronization is disabled, ARB 1 and ARB 2 frequencies can be set independently.

5.3.2 Common freq.

Common generated signal frequency, in Hertz. Only visible when synchronization is enabled.

5.3.3 Sample count

Common number of samples per period of generated signal. Calculated automatically, not user-configurable.

5.4 ARB TUI tab: ”Generator ARBX” menus

These menus contain all parameters specific to a given ARB X arbitrary generator.

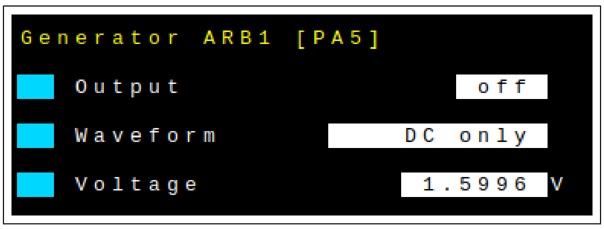


Figure 25: Generator parameters when the DC only waveform is selected

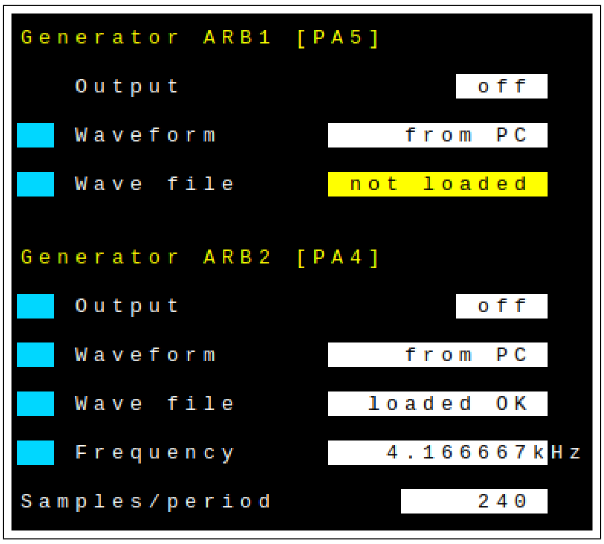


Figure 26: Generator parameters when using a custom waveform, before and after a wave file is loaded from the PC (ARB1, ARB2 respectively).

5.4.1 Output

- "off": Generator output is disabled. The GPIO output pin is left floating (high-Z state).
- "on": Generator output is enabled. The DAC drives the GPIO output pin.

5.4.2 Waveform

- "from PC": A custom waveform can be loaded from the PC.
- "DC only": Only a DC voltage is generated.
- "sinusoid": A sinusoid waveform is generated.
- "sawtooth": A sawtooth waveform with adjustable duty cycle is generated. Set duty cycle to 50 % to generate a symmetric triangle waveform.
- "square": A square waveform with adjustable duty cycle is generated.

5.4.3 Wave file

Only available when custom waveform is selected. Clicking the corresponding button opens a file selection dialog allowing the user to select a wave file to load from the PC. The value block shows the loading status, which can be one of the following:

- "not loaded": No file has been loaded since custom waveform was selected. The generator output cannot be enabled until a file is successfully loaded or the user selects a different waveform type. Highlighted in yellow to alert the user.
- "loading ... ": Wave file loading is in progress.
- "load error": An error occurred while loading the wave file. See Data Plotter's message log for the associated error message.

5.4.4 Frequency

Generated signal frequency, in Hertz. Only available when synchronization is disabled.

5.4.5 Phase

Initial signal phase, in degrees. Only available when synchronization is enabled. Hidden for custom waveform.

5.4.6 Duty cycle

Positive duty cycle of the generated signal, as a percentage of its period. Hidden for sinusoid, DC only and custom waveforms.

5.4.7 Amplitude

AC amplitude of the generated signal, in Volts. **Not** peak-to-peak value. Hidden for DC only and custom waveforms.

5.4.8 DC offset

DC offset of the generated signal, in Volts. Renamed to "Voltage" for DC only waveform. Hidden for custom waveform.

5.4.9 Sample count

Number of samples per period of generated signal. Calculated automatically, not user-configurable. Hidden when synchronization is enabled.

6 Frequency counter (CNT)

A frequency counter is also implemented. It can be used standalone or in conjunction with the oscilloscope in ETS mode, measuring its input signal frequency. The counter is based on two timers:

- The reference timer, clocked by the MCU timer clock (typically same frequency as system clock) to measure the time elapsed.
- The input timer, clocked by the input signal being measured by the counter.

The input signal frequency f_{IN} can be determined after the gate time has elapsed from the overall number of counted periods of both the reference (N_{REF}) and input (N_{IN}) timers by

$$f_{IN} = \frac{N_{IN}}{N_{REF}} \cdot f_{REF}, \tag{8}$$

where f_{REF} is the clock frequency of the reference timer. The nominal gate time set to 1 s, giving a worst-case frequency resolution of 1 Hz. This is also approximately the update rate of the counter. The counter always counts a whole number of input signal periods in order to maintain high accuracy for low-frequency input signals. This introduces a slight variance in the gate time, dependent on the input signal frequency. The frequency counter is always-on and cannot be disabled by the user. It only displays a number of measurements and system parameters, it has no configurable settings.

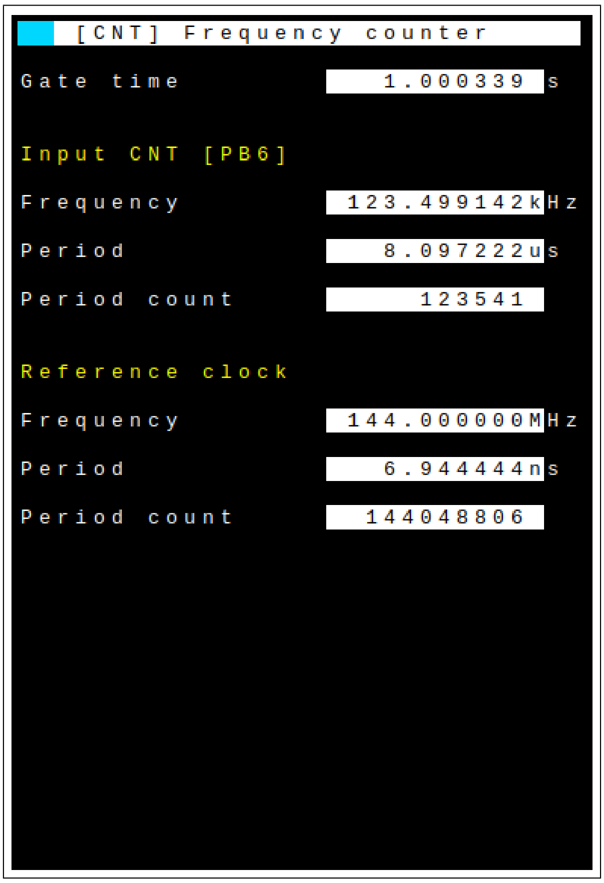


Figure 27: Frequency counter TUI tab

Note that due to the use of digital timer circuitry, the counter may not give correct readings for input signals which do not have clean, fast transitions between the logic low and high level voltages (e.g. slow or noisy analog signals). The counter may also count oscillations that are too fast to be captured by the oscilloscope.

6.1 CNT TUI tab: Gate time

The gate time, in seconds.

6.2 CNT TUI tab: "Input signal CNT" menu

6.2.1 Frequency

The measured frequency of the input signal, in Hertz.

6.2.2 Period

The measured period of the input signal, in seconds.

6.2.3 Period count

The number of input signal periods counted during the gate time.

6.3 CNT TUI tab: "Reference clock" menu

6.3.1 Frequency

The known frequency of the reference timer clock, in Hertz.

6.3.2 Period

The known period of the reference timer clock, in seconds.

6.3.3 Period count

The number of reference clock periods counted during the gate time.

7 Configuration profiles (PRO)

As the previous sections show, there is a large number of instrument parameters the user can configure. It would therefore be useful to make them persistent and avoid having to re-configure all of them manually every time the MCU is reset. To facilitate this, ”configuration profiles” containing all the parameters of every instrument are implemented. The current parameter values can be stored in a profile, while previously stored values can be recalled from one.

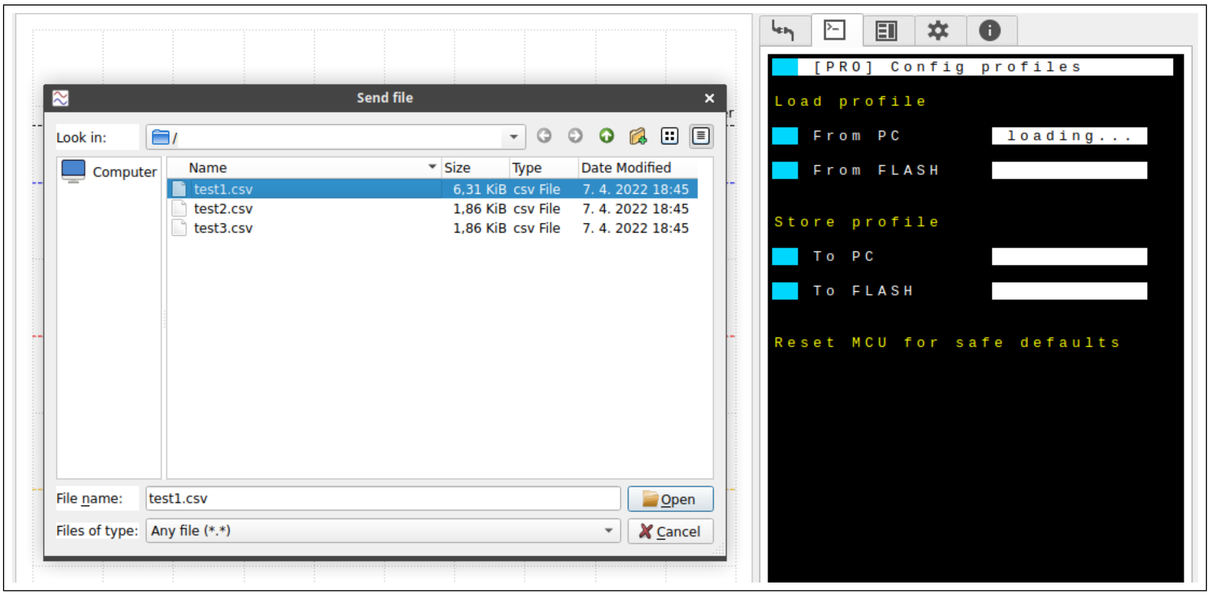


Figure 28: Configuration profiles TUI tab with open file selection dialog

A single profile can be stored in the MCU’s internal FLASH memory. It can then be loaded even after an MCU reset or reprogramming. Using Data Plotter, it is also possible to download/upload profiles to/from the connected PC. The profiles are stored as non-human-readable .csv files. Their size is approx. 500 kB + 3 B per sample of arbitrary generator waveform (if using custom waveform).

Each profile contains a firmware version descriptor to ensure compatibility. Therefore, the firmware will reject a profile saved from another MCU platform or firmware release.

7.1 PRO TUI tab: ”Load profile” menu

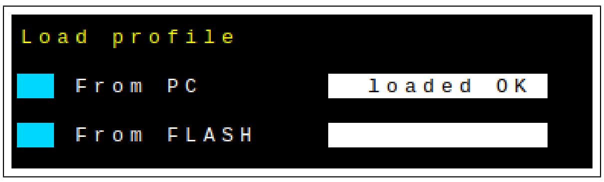


Figure 29: Load profile menu after a profile has been successfully loaded from PC

7.1.1 From PC

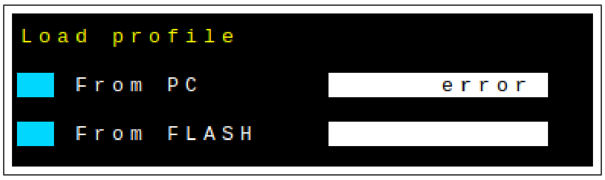
Loads instrument parameter values from profile stored in the PC, opening a file selection window.

7.1.2 From FLASH

Loads instrument parameter values from profile stored in the internal FLASH memory.

Value fields show the load operation status:

- (blank): Initial state (no profile loaded since last MCU reset).
- "loading ... ": Profile loading is in progress.
- "loaded OK": Profile has been loaded successfully.
- "error": An error occurred while loading profile. See Data Plotter's message log for the associated error message.



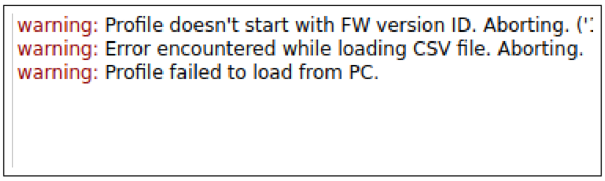


Figure 30: Load profile menu after an error occurred (left), corresponding error message (right)

7.2 PRO TUI tab: "Store profile" menu

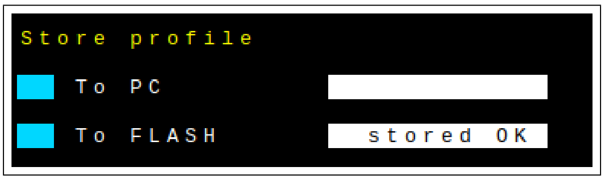


Figure 31: Store profile menu after a profile has been successfully stored to FLASH

7.2.1 To PC

Stores current instrument parameter values into a profile stored in the PC, opening a file selection window.

7.2.2 to FLASH

Stores current instrument parameter values into a profile stored in the internal FLASH memory.

Value fields show the store operation status:

- (blank): Initial state (no profile stored since last MCU reset).
- "storing ... ": Profile storing is in progress.
- "stored OK": Profile has been stored successfully.
- "error": An error occurred while storing profile. See Data Plotter's message log for the associated error message.

8 Versions of VSVI platform (supported MCUs)

8.1 ”F303-Nucleo64” for STM32F303RE on Nucleo-F303RE

The primary adaptation of the developed VSVI platform is for the Nucleo-F303RE development board shown in Figure 32. This board is based on the STM32F303RE microcontroller in the LQFP64 package and is one of the most widely used STM32 platforms at FEE CTU. It includes an ST-Link V2-1 debugger/programmer which can be used to program the MCU using the onboard Mini-USB connector. The ST-Link additionally serves as a USB-UART converter, being connected to the USART2 interface of the STM32F303RE MCU. The ST-Link clock derived from an onboard 8 MHz crystal oscillator is also used as the clock source for the STM32F303RE MCU in HSE bypass mode. Figure 33 shows the pinout of this version of the VSVI platform.

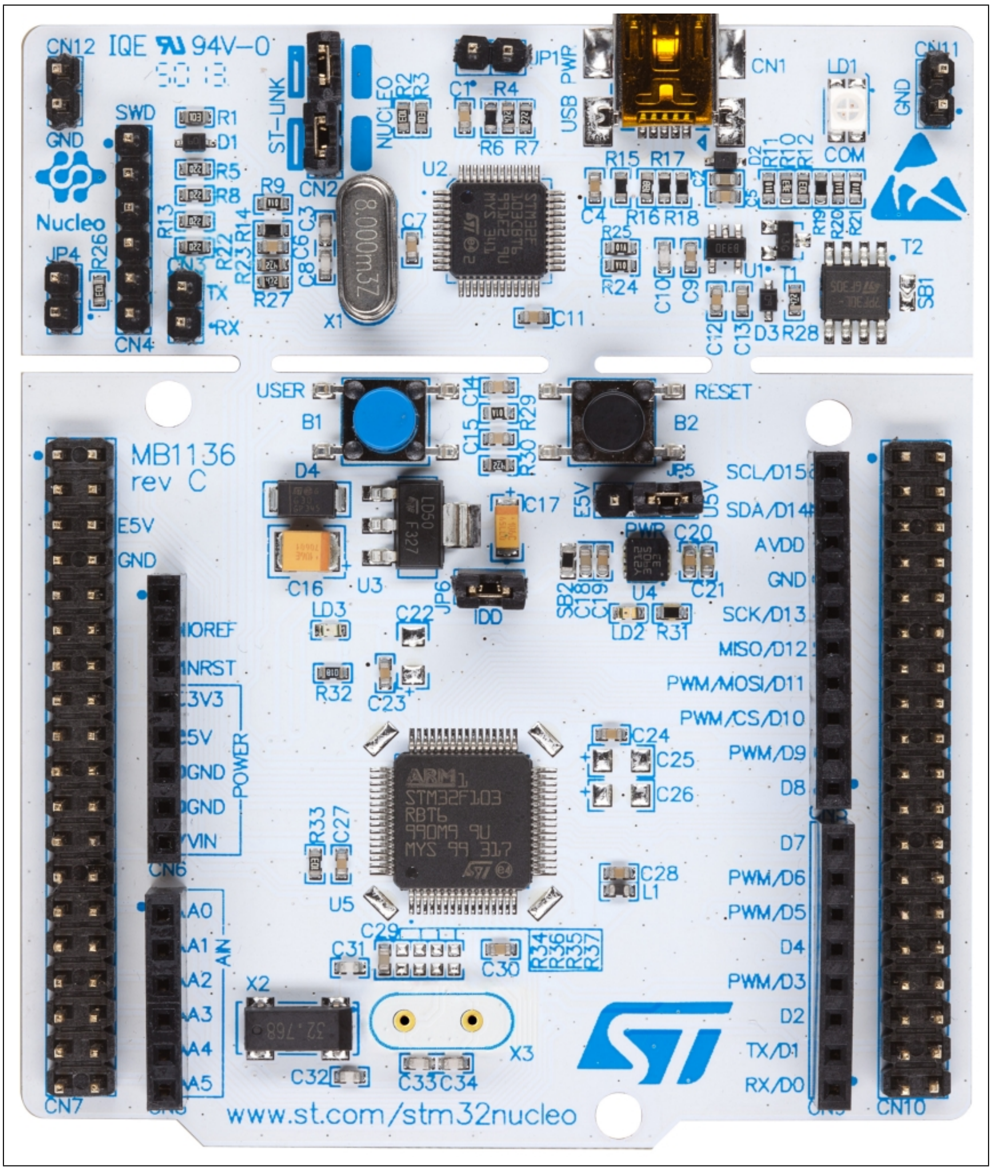


Figure 32: Nucleo-F303RE development board with STM32F303RE MCU

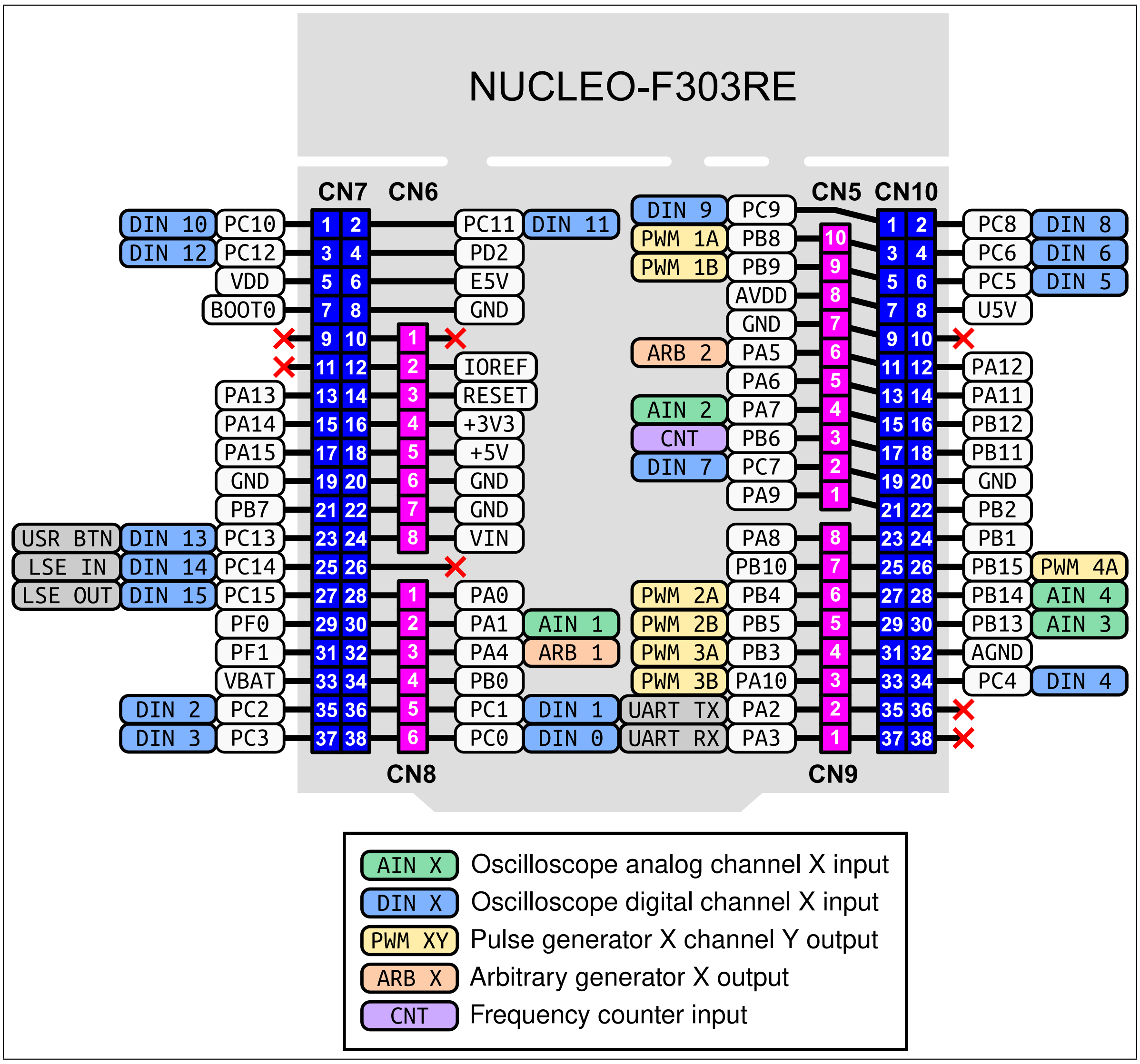


Figure 33: Pinout of SDIs developed for STM32F303RE (Nucleo-F303RE development board)

8.1.1 Mixed-signal oscilloscope (STM32F303RE)

- **4 analog channels** (inputs AIN 1, AIN 2, AIN 3, AIN 4) with **12-bit resolution**
- **16 digital channels** (inputs DIN 0 – DIN 15)
- Maximum real-time sampling rate:
  - **24 MSps** for all digital channels with analog channels disabled
  - **18 MSps** for one analog channel with 4 ADCs interleaving, digital channels disabled
  - **5 MSps** for all digital and analog channels with no interleaving

– Table 2 shows all the possible channel configurations and their respective max. sampling rates when all arbitrary generators are disabled. When any of the arbitrary generators is enabled, Table 3 applies instead.

- Maximum equivalent-time sampling rate:
  – Depends on the input signal frequency
  – Typically up to 72 MSps, max. 720 MSps with 1 sample per 10 input signal periods

Digital \ Analog	**Disabled**	**1 channel**	**2 channels**	**4 channels**
Disabled	-	18.0 MSps (i4) 10.3 MSps (i2) 5.1 MSps (-)	10.3 MSps (i2) 5.1 MSps (-)	5.1 MSps (-)
16 channels	24 MSps (-)	10.3 MSps (i2) 5.1 MSps (-)	10.3 MSps (i2) 5.1 MSps (-)	4.8 MSps (-)

Table 2: Maximum MSO sampling rates for STM32F303RE with arbitrary generators disabled. Used ADC interleaving mode indicated in parentheses ("-" = none, "i2" = 2 ADCs/channel, "i4" = 4 ADCs/channel).

Digital \ Analog	**Disabled**	**1 channel**	**2 channels**	**4 channels**
Disabled	-	7.2 MSps (i2) 4.8 MSps (-)	7.2 MSps (i2) 4.8 MSps (-)	4.0 MSps (-)
16 channels	8 MSps (-)	5.5 MSps (i2) 4 MSps (-)	5.5 MSps (i2) 4 MSps (-)	3.4 MSps (-)

Table 3: Maximum MSO sampling rates for STM32F303RE with arbitrary generators enabled. Used ADC interleaving mode indicated in parentheses ("-" = none, "i2" = 2 ADCs/channel).

- Maximum record length depends on the number of enabled channels according to Table 4

Digital \ Analog	**Disabled**	**1 channel**	**2 channels**	**4 channels**
Disabled	-	30.8 kSa	15.4 kSa	7.7 kSa
16 channels	30.8 kSa	15.4 kSa	10.2 KSa	6.1 kSa

Table 4: Maximum MSO record lengths for STM32F303RE

- **When interleaved sampling of the analog channels is used, the user must connect external jumper links between analog input pins:**
  – When 1 analog channel is enabled and interleaving 2 ADCs per channel, connect:
    * AIN 1 (PA1) and AIN 3 (PB13)

- When 2 analog channels are enabled and interleaving 2 ADCs per channel, connect:
  * AIN 1 (PA1) and AIN 3 (PB13)
  * AIN 2 (PA7) and AIN 4 (PB14)
- When 1 analog channel is enabled and interleaving 4 ADCs per channel, connect:
  * AIN 1 (PA1) and AIN 3 (PB13)
  * AIN 2 (PA7) and AIN 4 (PB14)
  * AIN 3 (PA13) and AIN 4 (PB14) or AIN 1 (PA1) and AIN 2 (PA7)

- **Some digital channels are limited on stock Nucleo-F303 boards:**
  - DIN 13 (PC13) is connected to the USER push button, including a 4.7 kΩ pull-up resistor to the 3.3V rail and a bypass capacitor. External signals should not be connected to this pin unless the solder bridge SB17 is removed, disconnecting the button circuitry.
  - DIN 14 and DIN 15 are not available. The corresponding MCU pins PC14 and PC15 are connected to the 32 kHz LSE crystal oscillator and NOT connected to the pin headers. To use these digital channels, R34 and R36 must be removed (disconnecting the crystal) and SB48 and SB49 must be connected (connecting the MCU pins to the pin headers).

8.1.2 Pulse generators (STM32F303RE)

- **4 generators** available:
  - PWM 1 with two channels (PWM 1A, PWM 1B)
  - PWM 2 with two channels (PWM 2A, PWM 2B)
  - PWM 3 with two channels (PWM 3A, PWM 3B)
  - PWM 4 with one channel (PWM 4A)
- All can run independently or be synchronized with adjustable phase/time delay
- Maximum output signal frequency 72 MHz (50% duty cycle only)

8.1.3 Arbitrary generators (STM32F303RE)

- **2 generators** available:
  - ARB 1
  - ARB 2
- Both can run independently or be synchronized with adjustable phases
- Maximum sampling frequency 1 MHz
- Maximum output signal frequency 250 kHz (4 samples per period)
- Up to 1000 samples per period for output signal frequencies $\leq$ 1 kHz

8.1.4 Frequency counter (STM32F303RE)

- Input signal frequency range: 10 Hz – 72 MHz
- Gate time $\approx$ 1 s gives update rate $\approx$ 1 Hz
- Frequency resolution $\leq$ 1 Hz using modified frequency ratio measurement (see section 6)

8.2 "G431-LQFP32" for STM32G431KB on LQFP32 adapter

The VSVI platform was also adapted for the STM32G431KB microcontroller in the LQFP32 package. The embedded USB peripheral is used for communication with the PC. An external 3.3 V voltage regulator is needed to supply the VDD supply voltage. An 8 MHz crystal oscillator should also be connected to the HSE IN, HSE OUT pins with the appropriate load capacitors. Using this crystal oscillator significantly improves clock stability and the frequency accuracy of the generators and frequency counter. It is also absolutely necessary for MSO equivalent-time sampling of externally-generated signals. In case no crystal oscillator is detected at MCU startup, the internal RC oscillator (HSI) is used instead. Figure 34 shows the pinout of the VSVI platform for this MCU, as soldered on an LQFP32-to-DIP adapter for use in breadboards.

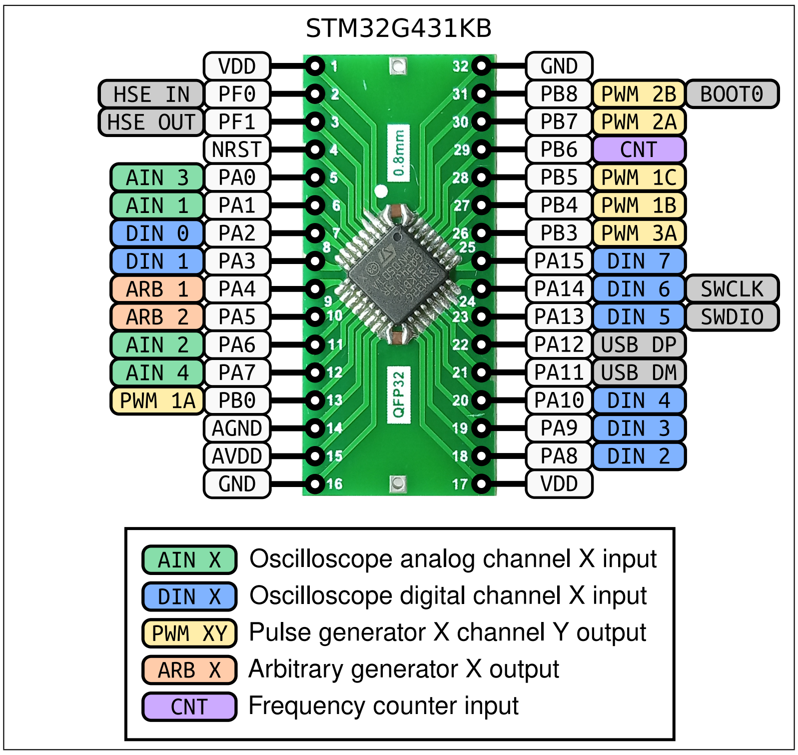


Figure 34: Pinout of SDIs developed for STM32G431KB (LQFP32-to-DIP adapter)

8.2.1 Mixed-signal oscilloscope (STM32G431KB)

- **4 analog channels** (inputs AIN 1, AIN 2, AIN 3, AIN 4) with **12-bit resolution**
- **8 digital channels** (inputs DIN 0 – DIN 7)
- Maximum real-time sampling rate:
  - **20.8 MSps** for all digital channels with analog channels disabled
  - **6.5 MSps** for one analog channel with 2 ADCs interleaving, digital channels disabled
  - **1.7 MSps** for all digital and analog channels with no interleaving
  - Table 5 shows all the possible channel configurations and their respective max. sampling rates when all arbitrary generators are disabled. When any of the arbitrary generators is enabled, Table 6 applies instead.
- Maximum equivalent-time sampling rate:
  - Depends on the input signal frequency
  - Typically up to 104 MSps, max. 1.04 GSa/s with 1 sample per 10 input signal periods

Analog / Digital	Disabled	1 channel	2 channels	4 channels
Disabled	-	6.5 MSps (i2) 3.4 MSps (-)	3.4 MSps (-)	1.7 MSps (-)
8 channels	20.8 MSps (-)	6.5 MSps (i2) 3.4 MSps (-)	3.4 MSps (-)	1.7 MSps (-)

Table 5: Maximum MSO sampling rates for STM32G431KB with arbitrary generators disabled. Used ADC interleaving mode indicated in parentheses ("-" = none, "i2" = 2 ADCs/channel).

Analog / Digital	Disabled	1 channel	2 channels	4 channels
Disabled	-	6.5 MSps (i2) 3.4 MSps (-)	3.4 MSps (-)	1.7 MSps (-)
8 channels	13.0 MSps (-)	6.5 MSps (i2) 3.4 MSps (-)	3.4 MSps (-)	1.7 MSps (-)

Table 6: Maximum MSO sampling rates for STM32G431KB with arbitrary generators enabled. Used ADC interleaving mode indicated in parentheses ("-" = none, "i2" = 2 ADCs/channel).

- Maximum record length depends on the number of enabled channels according to Table 7

Analog / Digital	Disabled	1 channel	2 channels	4 channels
Disabled	-	9.2 kSa	4.6 kSa	2.3 kSa
8 channels	9.2 kSa	4.6 kSa	3.1 KSa	1.8 kSa

Table 7: Maximum MSO record lengths for STM32G431KB

- **When interleaved sampling of analog channel 1 is used, the user must connect external jumper links between the following analog input pins:**
  - AIN 1 (PA1) and AIN 2 (PA6)

8.2.2 Pulse generators (STM32G431KB)

- **3 generators** available:
  - PWM 1 with three channels (PWM 1A, PWM 1B, PWM 1C)
  - PWM 2 with two channels (PWM 2A, PWM 2B)
  - PWM 3 with one channel (PWM 3A)
- All can run independently or be synchronized with adjustable phase/time delay
- Maximum output signal frequency 52 MHz (50% duty cycle only)

8.2.3 Arbitrary generators (STM32G431KB)

- **2 generators** available:
  - ARB 1
  - ARB 2
- Both can run independently or be synchronized with adjustable phases
- Maximum sampling frequency 1 MHz
- Maximum output signal frequency 250 kHz (4 samples per period)
- Up to 1000 samples per period for output signal frequencies $\leq$ 1 kHz

8.2.4 Frequency counter (STM32G431KB)

- Input signal frequency range: 10 Hz – 52 MHz
- Gate time $\approx$ 1 s gives update rate $\approx$ 1 Hz
- Frequency resolution $\leq$ 1 Hz using modified frequency ratio measurement (see section 6)

9 Known issues

9.1 Firmware freezes when configuration profile loading from PC is cancelled

Description

When the user cancels the file selection dialog, no data is received by the MCU. The firmware then keeps waiting to receive data indefinitely, appearing frozen. Since it is impossible to distinguish between TUI commands and file contents sent from the PC, the firmware cannot recover from this condition and an MCU reset is needed.

This does not apply when storing a profile to PC. In that case, the data is transferred to the PC first, before the file selection dialog is shown to the user. If the user cancels it, the data is discarded by Data Plotter.

Encountered in Data Plotter **release 17.3.2022 and earlier**, will be fixed in the next release.

Workaround

Do not cancel the file selection dialog when loading a profile from PC. Otherwise, reset MCU to recover.

9.2 TUI appears truncated after Data Plotter window is enlarged

Description

When the Data Plotter window is resized, the terminal window expands in height but its active area where the TUI is displayed does not. The TUI then appears truncated.

Encountered in Data Plotter **release 17.3.2022 and earlier**, will be fixed in the next release.

Workaround

It is still possible to use the TUI and access the hidden elements by scrolling. However, to fully use the new height of the terminal window, a new MCU connection must be made. Therefore, it's necessary to disconnect from the MCU, reset the MCU and reconnect.