IQTools – Waveform Generation Tool

For Keysight Arbitrary Waveform Generators

KEYSIGHT

USER GUIDE

Notices

Trademarks

PCI Express® and PCIe® are registered trademarks of PCI-SIG.

Manual Part Number

M8100-91B60

Edition

Edition 7.0, November 2025

Keysight Technologies Deutschland GmbH
Herrenberger Strasse 130,
71034 Böblingen, Germany

Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

U.S. Government Rights

The Software is “commercial computer software,” as defined by Federal Acquisition Regulation (“FAR”) 2.101. Pursuant to FAR 12.212 and 27.405-3 and Department of Defense FAR Supplement

(“DFARS”) 227.7202, the U.S. government acquires commercial computer software under the same terms by which the software is customarily provided to the public. Accordingly, Keysight provides the Software to U.S. government customers under its standard commercial license, which is embodied in its End User License Agreement (EULA), a copy of which can be found at http://www.keysight.com/find/sweula. The license set forth in the EULA represents the exclusive authority by which the U.S. government may use, modify, distribute, or disclose the Software. The EULA and the license set forth therein, does not require or permit, among other things, that Keysight: (1) Furnish technical information related to commercial computer software or commercial computer software documentation that is not customarily provided to the public; or (2) Relinquish to, or otherwise provide, the government rights in excess of these rights customarily provided to the public to use, modify, reproduce, release, perform, display, or disclose commercial computer software or commercial computer software documentation. No additional government requirements beyond those set forth in the EULA shall apply, except to the extent that those terms, rights, or licenses are explicitly required from all providers of commercial computer software pursuant to the FAR and the DFARS and are set forth specifically in writing elsewhere in the EULA. Keysight shall be under no obligation to update, revise or otherwise modify the Software. With respect to any technical data as defined by FAR 2.101, pursuant to FAR 12.211 and 27.404.2 and DFARS 227.7102, the U.S. government acquires no greater than Limited Rights as defined in FAR 27.401 or DFAR 227.7103-5 (c), as applicable in any technical data.

Warranty

THE MATERIAL CONTAINED IN THIS DOCUMENT IS PROVIDED "AS IS," AND IS SUBJECT TO BEING CHANGED, WITHOUT NOTICE, IN FUTURE EDITIONS. FURTHER, TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, KEYSIGHT DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED WITH REGARD TO THIS MANUAL AND ANY INFORMATION CONTAINED HEREIN, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. KEYSIGHT SHALL NOT BE LIABLE FOR ERRORS OR FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, USE, OR PERFORMANCE OF THIS DOCUMENT OR ANY INFORMATION CONTAINED HEREIN. SHOULD KEYSIGHT AND THE USER HAVE A SEPARATE WRITTEN AGREEMENT WITH WARRANTY TERMS COVERING THE MATERIAL IN THIS DOCUMENT THAT CONFLICT WITH THESE TERMS, THE WARRANTY TERMS IN THE SEPARATE AGREEMENT WILL CONTROL.

Safety Notices

CAUTION

A CAUTION notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a CAUTION notice until the indicated conditions are fully understood and met.

WARNING

A WARNING notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a WARNING notice until the indicated conditions are fully understood and met.

Keysight Cybersecurity

Product and Solution Cybersecurity

Keysight is dedicated to ensuring the cybersecurity of its products and solutions. For detailed information, visit Product and Solution Cybersecurity.

Report a Product Cybersecurity Issue

If you encounter a cybersecurity issue with a Keysight product, report it immediately at: Product Cybersecurity Issue Reporting.

Responsible Disclosure Program

Keysight encourages responsible disclosure of security vulnerabilities. For more details, visit Responsible Disclosure Program.

Product Software Updates

Keysight releases periodic software updates to fix known defects, incorporate product enhancements, and address cybersecurity vulnerabilities, if any. Ensure that your product software is always up to date. To search for software updates for your product, visit the Keysight Technical Support page at: https://www.keysight.com/find/iqtools.

User Documentation

User documentation includes comprehensive information on features that may impact the secure deployment, use, and decommissioning of our products. This includes details on exposed network ports, the use of cryptography and authentication, and firmware security settings, if necessary.

Safety Summary

The following general safety precautions must be observed during all phases of operation of this instrument. Failure to comply with these precautions or with specific warnings or operating instructions in the product manuals violates safety standards of design, manufacture, and intended use of the instrument. Keysight Technologies assumes no liability for the customer's failure to comply with these requirements. Product manuals are provided with your instrument on CD-ROM and/or in printed form. Printed manuals are an option for many products. Manuals may also be available on the Web. Go to www.keysight.com and type in your product number in the Search field at the top of the page.

General

This product is a Safety Class 1 instrument (provided with a protective earth terminal). The protective features of this product may be impaired if it is used in a manner not specified in the operation instructions.

All Light Emitting Diodes (LEDs) used in this product are Class 1 LEDs as per IEC 60825-1.

Environment Conditions

This instrument is intended for indoor use in an overvoltage category II, pollution degree 2 environment. It is designed to operate at a maximum relative humidity of 95% and at altitudes of up to 2000 meters.

Refer to the specifications tables for the ac mains voltage requirements and ambient operating temperature range.

Before Applying Power

Verify that all safety precautions are taken. The power cable inlet of the instrument serves as a device to disconnect from the mains in case of hazard. The instrument must be positioned so that the operator can easily access the power cable inlet. When the instrument is rack mounted the rack must be provided with an easily accessible mains switch.

Ground the Instrument

To minimize shock hazard, the instrument chassis and cover must be connected to an electrical protective earth ground. The instrument must be connected to the ac power mains through a grounded power cable, with the ground wire firmly connected to an electrical ground (safety ground) at the power outlet. Any interruption of the protective (grounding) conductor or disconnection of the protective earth terminal will cause a potential shock hazard that could result in personal injury.

Do Not Operate in an Explosive Atmosphere

Do not operate the instrument in the presence of flammable gases or fumes.

Do Not Remove the Instrument Cover

Operating personnel must not remove instrument covers. Component replacement and internal adjustments must be made only by qualified personnel.

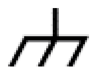
Instruments that appear damaged or defective should be made inoperative and secured against unintended operation until they can be repaired by qualified service personnel.

External Connections

Any other instruments connected to this instrument shall be approved to a suitable safety standard and must include reinforced insulation from hazardous voltages, in particular mains.

Safety Symbols

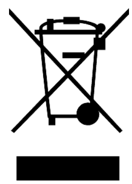
Table 1 Safety Symbols

Symbol	Description
	Indicates warning or caution. If you see this symbol on a product, you must refer to the manuals for specific Warning or Caution information to avoid personal injury or damage to the product.
	Frame or chassis ground terminal. Typically connects to the equipment's metal frame.
	KC is the Korean certification mark to demonstrate that the equipment is Class A suitable for professional use and is for use in electromagnetic environments outside of the home.
	Contains parts or assemblies susceptible to damage by electrostatic discharge (ESD). Use electrostatic discharge protective handling procedures to avoid malfunctions or potential damage to the instruments.
	Indicates the time period during which no hazardous or toxic substance elements are expected to leak or deteriorate during normal use. Forty years is the expected useful life of the product.
	The RCM Mark is a compliance mark to the ACMA (Australian Spectrum Management Agency). This indicates compliance with all Australian EMC regulatory information.
	Indicates that the product was tested and has met the certification requirements for electrical, plumbing and/or mechanical products.

Symbol	Description
CE CAN ICES/NMB-001(A) ISM GRP 1-A	The CE mark is a registered trademark of the European Community. This CE mark shows that the product complies with all the relevant European Legal Directives. CAN ICES/NMB-001(A) - This ISM device complies with the Canadian ICES-001(A). Cet appareil ISM est conforme a la norme NMB-001(A) du Canada. ISM GRP 1-A - This is an Industrial Scientific and Medical (ISM) Group 1 Class A product.
	This symbol on all primary and secondary packaging indicates compliance to China standard GB 18455-2001.

Compliance and Environmental Information

Table 2 Compliance and Environmental Information

Safety Symbol	Description
	The crossed out wheeled bin symbol indicates that separate collection for waste electric and electronic equipment (WEEE) is required, as obligated by DIRECTIVE 2012/19/EU and other National legislation. See http://about.keysight.com/en/companyinfo/environment/takeback.shtml to understand your Trade in options with Keysight in addition to product takeback instructions.

Contents

1 Introduction

KEYSIGHT TECHNOLOGIES

Overview

IQTools is a collection of MATLAB example applications for creating I/Q, IF/RF, serial data, multi-tone, radar pulse, and many other types of waveforms on the Keysight M8190A, M8194A, M8195A, M8196A, M8198A, M8199A, M8199B, 81180A, M933xA, 81150A, and 81160A arbitrary waveform generators (AWG) as well as several Keysight Signal Generators. The example applications are created to demonstrate the waveform generation capabilities of these instruments, along with the value of using these instruments together with MATLAB software.

MATLAB is a software environment and programming language that is very well suited for calculating arbitrary waveforms, measurement and analysis routines, and instrument applications. MATLAB is available directly from Keysight as an instrument option (that is, option N6171A-M03) with many instruments, including the M8190A and M933xA. It is also available from MathWorks.

IQTools requires the MATLAB N6171A-M03 package to operate (or the equivalent of these options, including MATLAB, Instrument Control Toolbox, Signal Processing Toolbox, Communications System Toolbox, and RF toolbox). These example applications were tested with MATLAB Version 9.7.0 (R2019b) and Version 9.8.0 (R2020a) and might not work on older versions of MATLAB.

Some of the analysis functionality works in conjunction with the PathWave VSA software (89600). Version 21.0 or higher is required. In this document, the PathWave 89600 VSA software is referred to as “VSA” software.

Visit the following URL to download the IQTools software:
www.keysight.com/find/IQTools

Limited Warranty

IQTools is provided “as-is” at no charge. The IQTools software has not completed Keysight’s full quality assurance program and may have errors or defects. Keysight makes no express or implied warranty of any kind with respect to the software, and specifically disclaims the implied warranties of merchantability and fitness for a particular purpose.

IQTools may only be used in conjunction with Keysight instruments.

Download IQTools

There are two flavors of **Download IQTools** available at Keysight IQTools web page: www.keysight.com/find/IQTools

Download IQTools (Requires MATLAB)

This download is provided as a zip file which contains source code to allow for customization and extension of IQTools functionality. Some routines are encrypted (*.p files) to protect Keysight's intellectual property.

This download requires MATLAB to operate. You can request a free MATLAB trial from your Keysight representative or at www.mathworks.com/keysight/trial.

Setup Instructions:

Download and extract the zip file into an empty directory, ensuring the folder structure remains intact.

Recommended location: A subdirectory of your MATLAB working directory (for example, *C:\Users\<username>\Documents\MATLAB\IQTools*).

Download IQTools (Standalone Executable)

This download is provided as an .exe file and does not require a MATLAB license. However, it does not support customization or extension of IQTools functionality.

Install IQTools

Follow these steps to install **IQTools (Standalone Executable)** on your system:

NOTE

Ensure that any previous version of IQTools is not installed on your system. If you find any, uninstall it through the Control Panel by navigating to 'Add or Remove Programs'. For uninstallation instructions, see "Uninstall IQTools" on page 25.

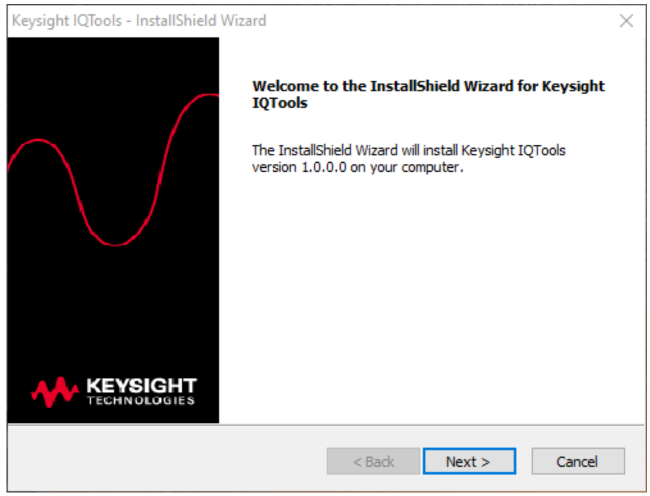
1 Go to Keysight IQTools web page: www.keysight.com/find/IQTools
2 Click **Download IQTools (Standalone Executable)** button to download the exe file.

NOTE

Verify your account permissions. Ensure that you have full administrative privileges (run as Administrator) before you install or upgrade the IQTools software on a PC running Windows 10 or 11. Not doing so may result in the installation failure. Please contact your system administrator to provide you the administrative rights.

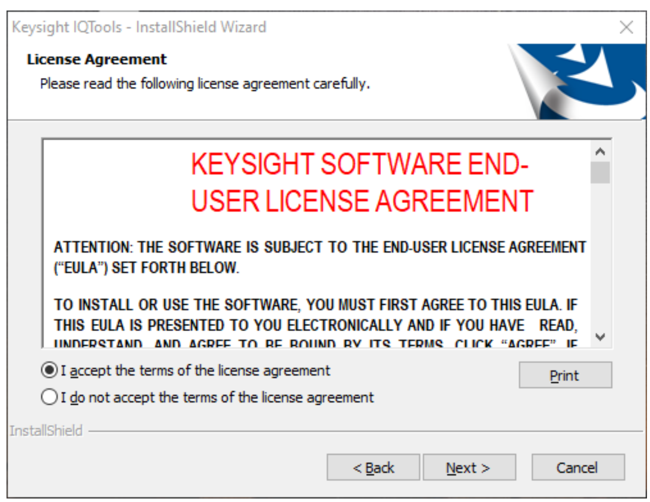
3 Double-click the exe file.

4 The **Keysight IQTools - InstallShield Wizard** screen appears.



Click **Next**.

5 The **Keysight License Agreement** screen appears.

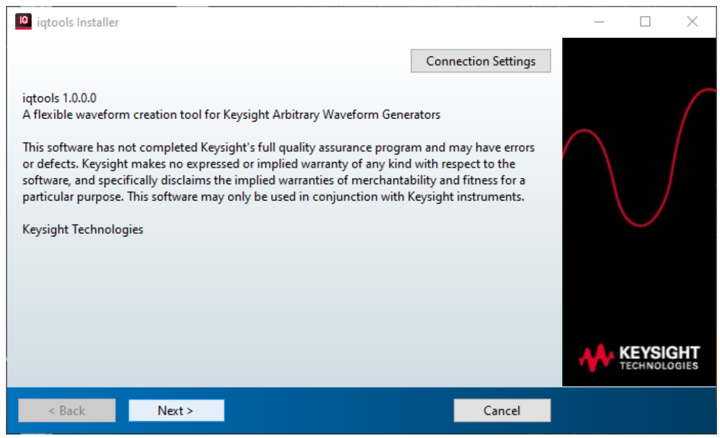


Select **I accept the terms of the license agreement** and click **Next**.

NOTE

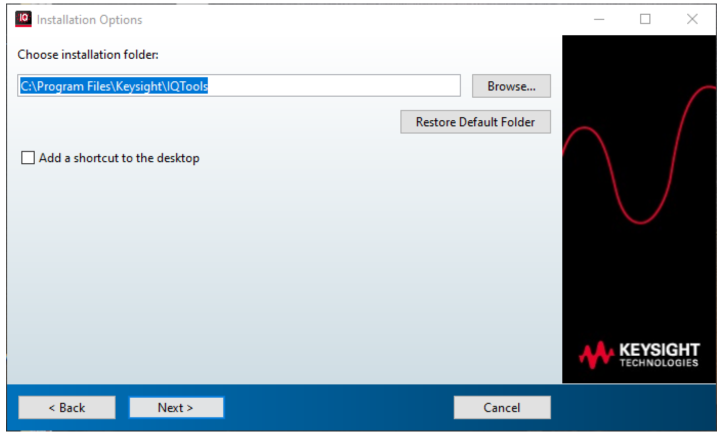
Note that the InstallShield Wizard may take some time to download the necessary files in the background. We recommend to wait until the next screen appears.

6 The **iqtools Installer** screen appears.



Click **Next**.

7 The **Installation Options** screen appears.



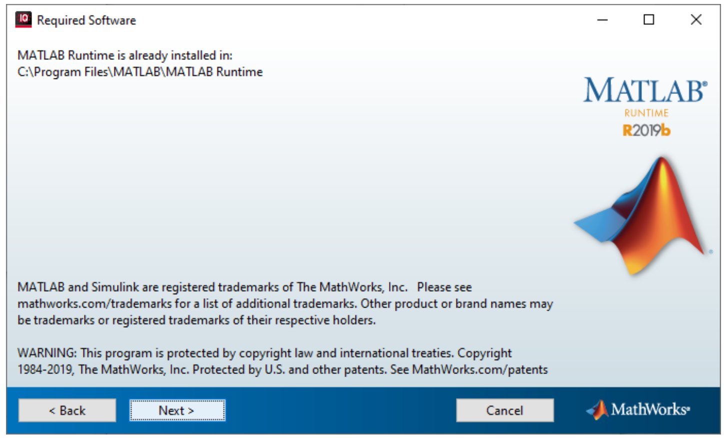
a (Optional) Click **Browse** to select the installation folder. By default, the location **C:\Program Files\Keysight\IQTools** is selected.

b (Optional) Select the **Add a shortcut to the desktop** check-box to create a shortcut on the desktop.

c Click **Next**.

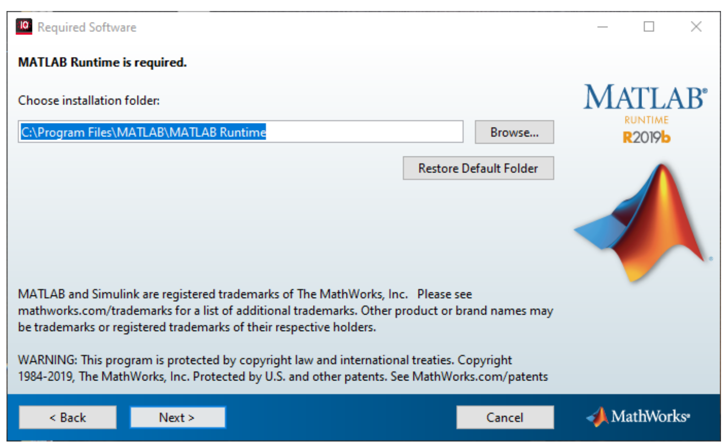
8 The **Required Software** screen appears based on the status of the MATLAB Runtime installation on your PC.

a If the MATLAB Runtime is already installed on your PC, the following **Required Software** screen appears.



i Click **Next**.

b Else, if the MATLAB Runtime is not installed on your PC, the following **Required Software** screen appears.

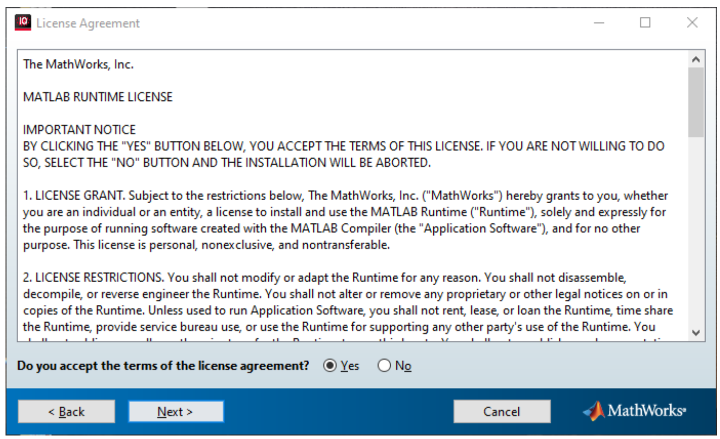


NOTE The MATLAB Runtime is already included in the exe file. It will not be downloaded separately.

i (Optional) Click **Browse** to select the installation folder. By default, the location **C:\Program Files\MATLAB\MATLAB Runtime** is selected.

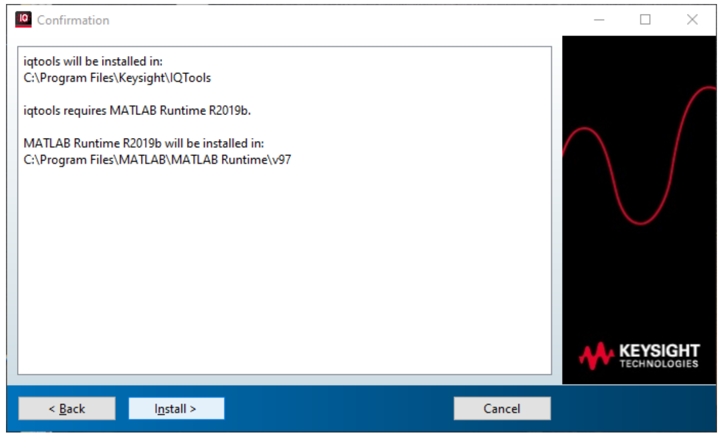
ii Click **Next**.

9 The **MATLAB License Agreement** screen appears.



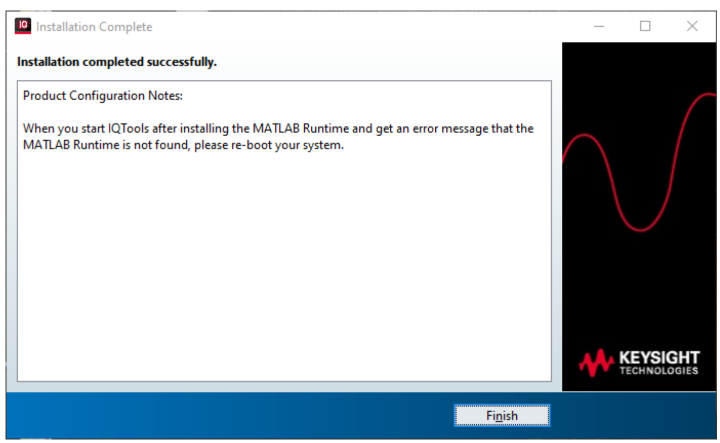
Select **Yes** and then click **Next**.

10 The **Confirmation** screen appears.



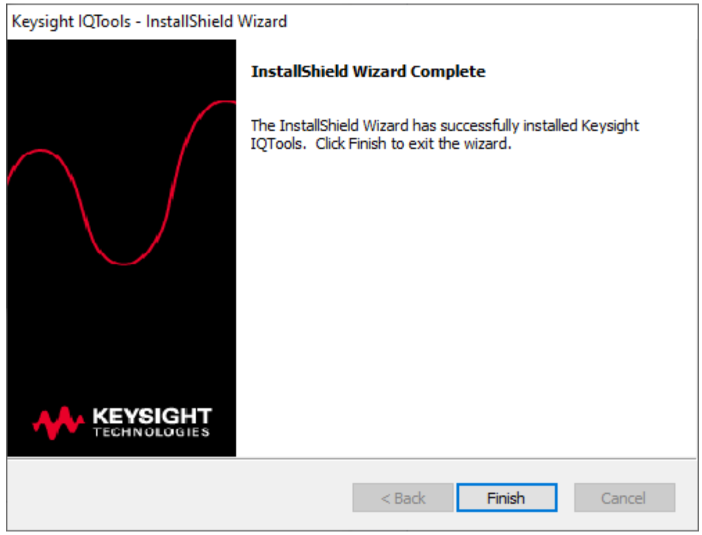
Click **Install** to start the installation process.

11 The **Installation Complete** screen appears after the installation process is over.



Click **Finish**.

12 The **InstallShield Wizard Complete** screen appears.



Click **Finish**.

This completes the installation process of **IQTools (Standalone Executable)**.

Uninstall IQTools

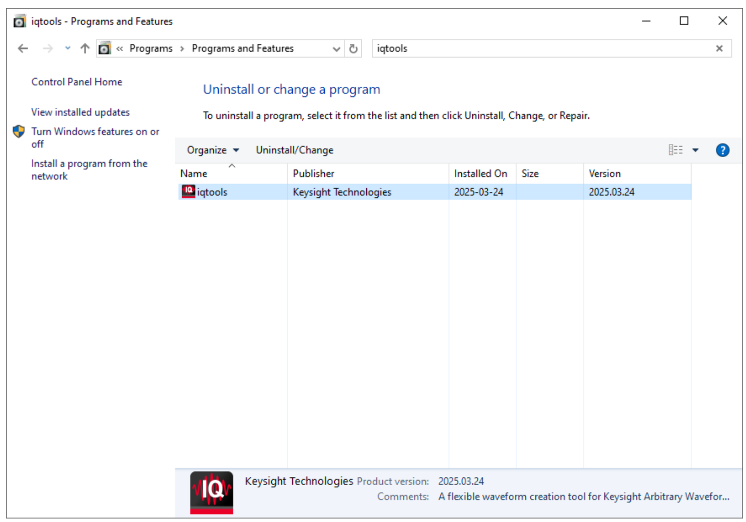
Follow these steps to uninstall **IQTools (Standalone Executable)** from your system:

1 Open the **Start** menu and search for **Control Panel**.

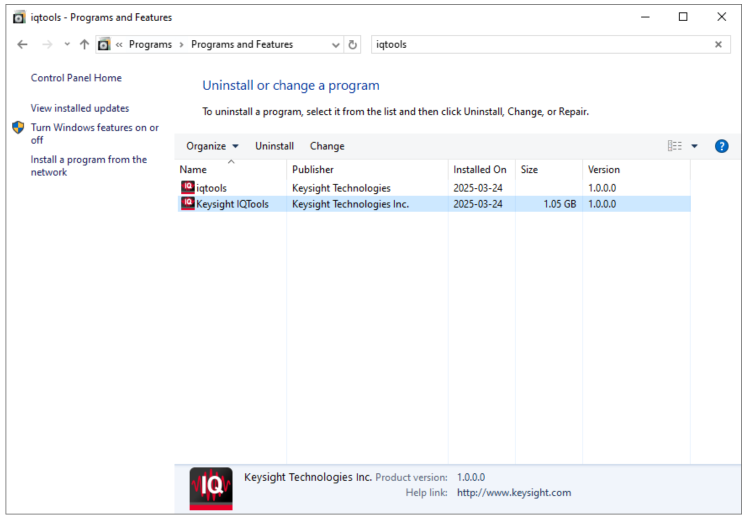
2 In the **Control Panel**, click on **Programs**, then **Programs and Features**. Alternatively, you can search directly for **Installed Apps** on your system.

3 Find **iqtools** in the list.

a If there is only one instance, **iqtools**; select it and click **Uninstall/Change**.



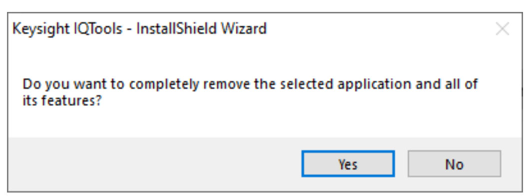
b If there are two instances, **iqtools** and **Keysight IQTools**; select **Keysight IQTools** and click **Uninstall**. It will also uninstall **iqtools** automatically.



NOTE

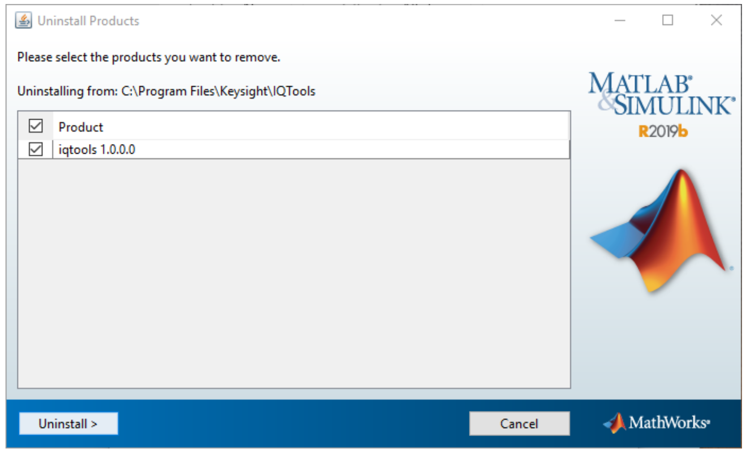
If there are two instances, **iqtools** and **Keysight IQTools**, and you uninstall **iqtools**, it will not uninstall **Keysight IQTools** automatically. In that case, you also need to uninstall **Keysight IQTools** manually by selecting it and clicking **Uninstall**.

The following message appears.



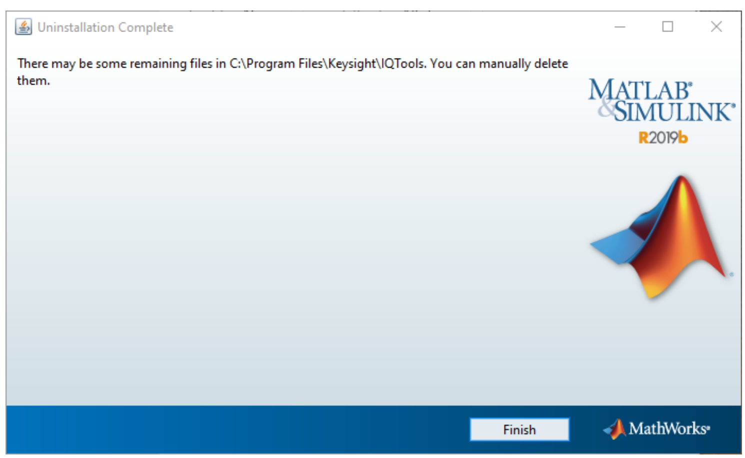
Click **Yes**.

4 The **Uninstall Products** screen appears.



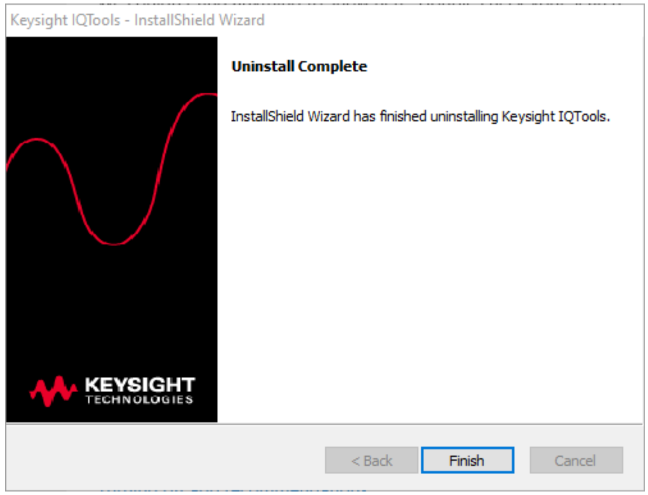
Select **Product** to select all the products (for example, iqtools 1.0.0.0) in the product list and click **Uninstall**.

5 The **Uninstallation Complete** screen appears.



Click **Finish**.

6 The **Uninstall Complete** screen appears.



Click **Finish**.

This completes the uninstallation process of **IQTools (Standalone Executable)**.

NOTE

After uninstallation, it is recommended to make sure that there is not any instance of **iqtools** left in the list at **Control Panel** > **Programs** > **Programs and Features**. Probably, you need to update the list by pressing F5.

Locate User Documentation

It includes the following documents:

- IQToolsUserGuide.pdf
- IQToolsReadMe.pdf
- IQToolsReleaseNotes.pdf
- IQToolsServerCommands.html
- KeysightEULA.pdf

For IQTools (Requires MATLAB)

The user documentation can be found in the “Documentation” folder available under the directory where you extracted IQTools zip file (for example, *C:\Users\<username>\Documents\MATLAB\IQTools*).

For IQTools (Standalone Executable)

The user documentation can be found in the “application” folder available under the directory where you installed IQTools (for example, *C:\Program Files\Keysight\IQTools)*.

Alternatively, you can also visit www.keysight.com/find/IQTools to find the IQTools User Guide and Release Notes.

Contact Keysight Service and Support

To locate a sales or a service office near you, go to

www.keysight.com/find/contactus

2 IQTools User Interface

KEYSIGHT TECHNOLOGIES

Launch IQTools User Interface

Launch IQTools (Requires MATLAB)

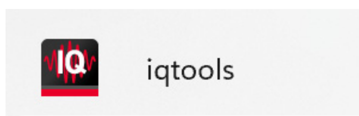
To launch IQTools easily, add the directory where you extracted the IQTools zip file (for example, *C:\Users\<username>\Documents\MATLAB\IQTools*) to MATLAB's search path and add the following line to your 'startup.m' script:

```
addpath('C:\Users\<username>\Documents\MATLAB\IQTools');
```

Type "iqtools" in the MATLAB command window and press Enter.

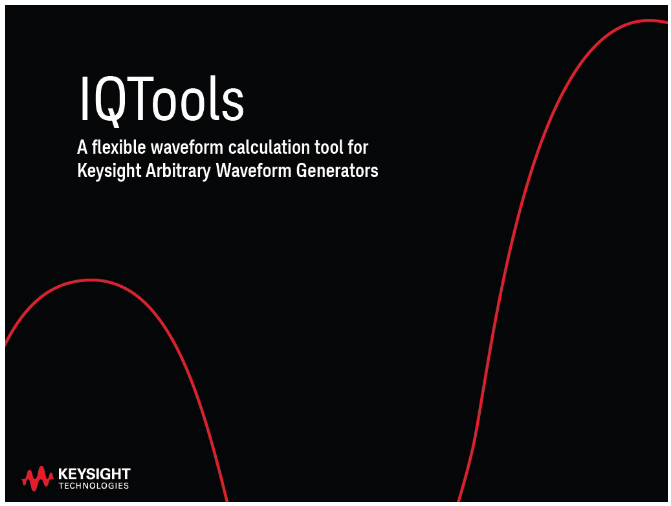
Launch IQTools (Standalone Executable)

Open **Start** menu and then write or search for **iqtools** and open it.



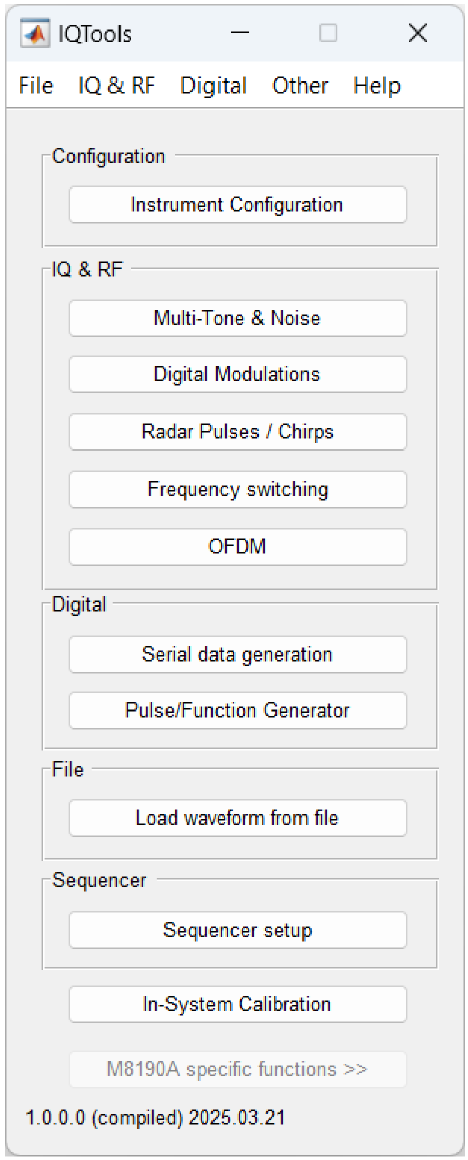
IQTools Splash Screen

After launching IQTools, the **Keysight IQTools Splash Screen** appears.



IQTools Main Window

After splash screen, the **Keysight IQTools Main Window** appears.



Version Information

At the bottom left of the main window, you can find the version of IQTools, along with the build type (either “script” or “compiled”) and the build date (YYYY.MM.DD). The “script” type refers to the **IQTools (Requires MATLAB)**, while the “compiled” type refers to the **IQTools (Standalone Executable)** (see “Download IQTools” on page 16).

In case you need support, include this version information.

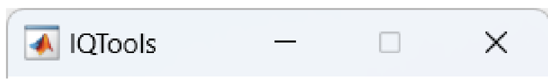
GUI Elements

The IQTools main window consists of the following GUI elements:

- Title Bar
- Menu Bar
- Button Panel

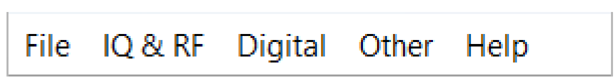
Title Bar

The title bar contains an application icon, title and standard buttons to minimize, maximize or to close the window.



Menu Bar

The menu bar consists of various drop-down menus which provide access to different functions, and launch interactive GUI controls.



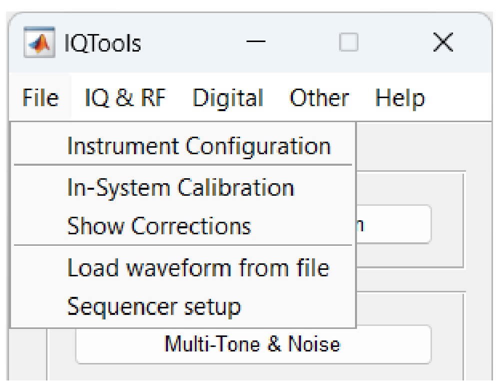
The menu bar includes the following drop-down menus:

- File
- IQ & RF
- Digital
- Other
- Help

File Menu

The **File** menu provides the following selection:

- **Instrument Configuration:** Enables you to select the AWG or signal generator model from the list available and connect IQTools software with the AWG, signal generator, oscilloscope, and so on, by providing their visa addresses. For more information, see “Instrument Configuration” on page 64.
- **In-System Calibration**: Allows you to capture the magnitude and phase response of your AWG or signal generator. This data is used to pre-distort a waveform to achieve a flat response. For more information, see “In-System Calibration” on page 131.
- **Show Corrections**: Clicking this opens the Correction Management window. For more information, see “Correction Management” on page 146.
- **Load waveform from file**: This allows you to feed external data to IQTools by loading a waveform file. For more information, see “Load Waveform from File” on page 114.
- **Sequencer setup**: This is specific to M8190A, M8195A and M8198A. All the AWGs that support sequencing allow you to play sequential data by entering the required fields and defining the sequence as per your requirement. For more information, see “Sequencer Setup” on page 120.

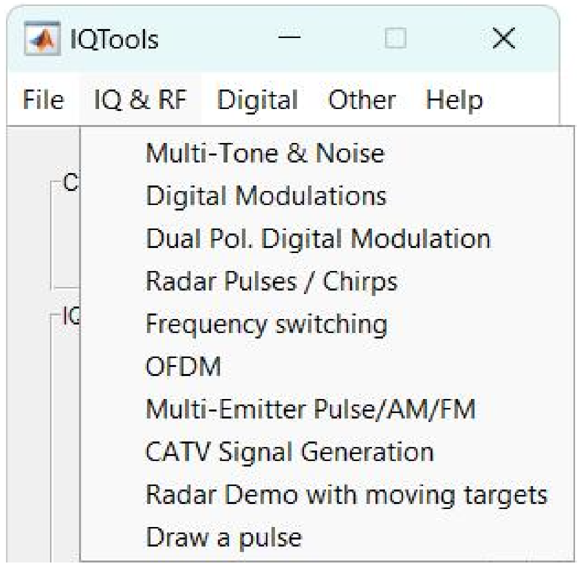


IQ & RF Menu

The IQ & RF menu offers a number of frequency-domain oriented waveform generation utilities. It provides the following selection:

- **Multi-Tone & Noise**: Allows you to create signals made up of one or more tones, either equally or arbitrarily spaced. It also allows for the definition of a frequency interval without tones (or notch) for NPR (Noise Power Ratio) testing. Amplitudes and phases of the individual tones can be corrected through correction factor files that you have defined. The Multi-Tone tab allows you to generate both RF and baseband (I/Q) signals. For more information, see “Multi-Tone and Noise” on page 71.
- **Digital Modulations**: Allows you to create digitally modulated signals, both at IF/RF and baseband (I/Q). It directly supports a large variety of signal-carrier modulation schemes. For more information, see “Digital Modulations” on page 77.
- **Dual Polarization Digital Modulations**: Allows you to create dual polarization digitally modulated signals. For more information, see “Dual Polarization Digital Modulation” on page 83.
- **Radar Pulses/Chirps**: Use this option to create a variety of pulsed signals. It allows you to generate both RF and Baseband (I/Q) signals. It directly supports a variety of pulse train arrangements and intra-modulation schemes. For more information, see “Radar Pulses/Chirps” on page 89.
- **Frequency switching**: Allows frequency switching between two or more frequencies with adjustable tone duration and phase continuous switching. For more information, see “Frequency Switching” on page 96.
- **OFDM**: This option enables signal generation with orthogonal frequency division multiplex modulation scheme. For more information, see “OFDM” on page 97.
- **Multi-Emitter Pulse/AM/FM**: This option allows you to generate overlapping pulse sequences along with multiple AM or FM modulated carriers.
- **CATV Signal Generation**: Generate a waveform that simulates a cable TV signal with multiple carriers.
- **Radar Demo with moving targets**: Using the sequencer capability of the M8190A, M8195A or M8198A AWGs, this option allows you to generate a signal scenario of moving radar targets.

- **Draw a pulse**: This option allows you to generate a pulse by drawing the magnitude and other defining parameters of a pulse on the graph. You can select which parameter to draw/define using the 'Creation Mode' drop-down menu.

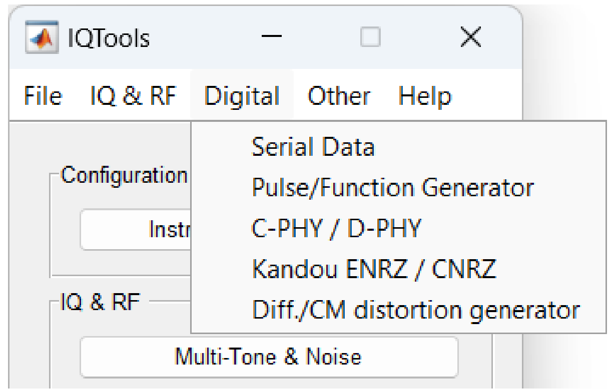


Digital Menu

The **Digital** menu contains a number of waveform generation utilities for time-domain signals. It provides the following selection:

- **Serial Data**: Use this option to create single lane and multi lane bi-level and multi-level high-speed digital serial signals, both with and without distortions. It supports a variety of channel coding and modulation schemes and allows you to generate a clock signal on a different AWG channel or marker output. For more information, see "Serial Data Generation" on page 98.
- **Pulse/Function Generator**: Allows you to generate time-domain pulses or pulse sequences by defining the pulse parameters.
- **C-PHY/D-PHY**: Generate C-PHY and D-PHY signals using this option.
- **Kandou ENRZ/CNRZ**: Generate ENRZ/CNRZ signals using this option.

- **Diff./CM distortion generator**: Use this option to use the M8195A AWG as a differential & common mode distortion generator in conjunction with another instrument, such as a BERT pattern generator.



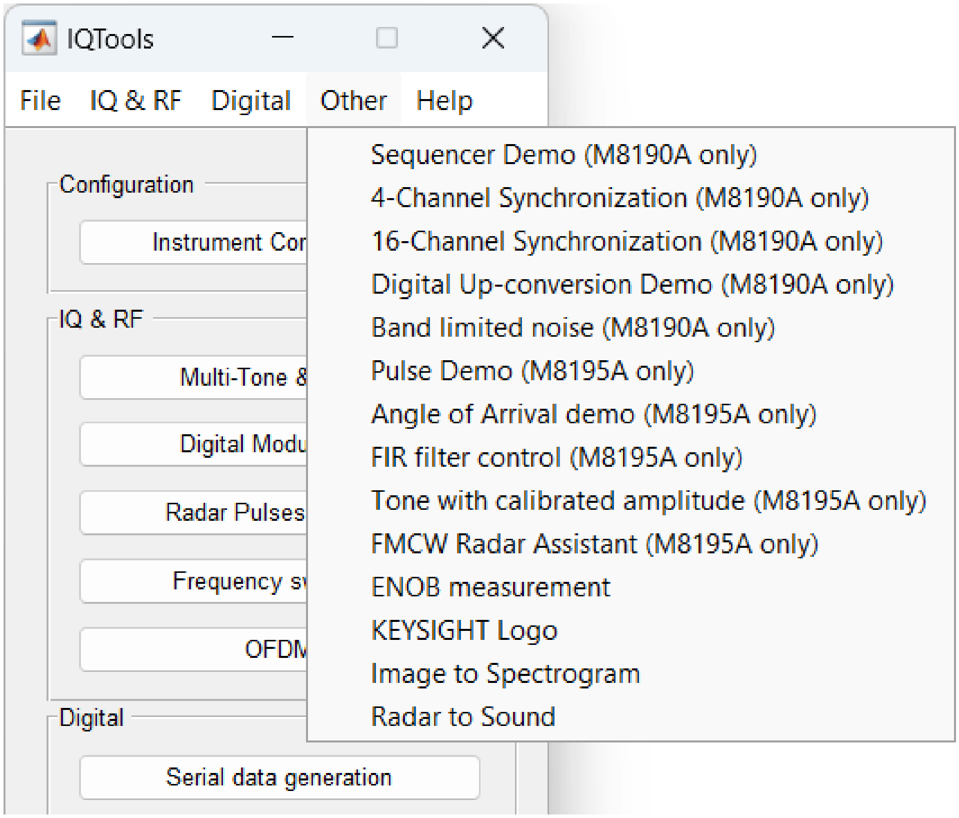
Other Menu

This menu contains a collection of example applications. Some of them are “pushbutton demos”, which can be used to demonstrate a particular AWG feature. Others are specialized measurements that work only in conjunction with certain external equipment and finally, some are simply “fun” applications, such as the “Keysight Logo” shown on an oscilloscope.

The **Other** menu provides the following selection:

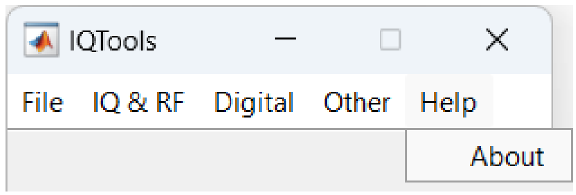
- **Sequencer Demo (M8190A only)**: This example demonstrates the M8190A’s capability for static and dynamic sequencing as well as memory ping-pong and shows how these functions can be programmed through IQTools. This utility works best with the script version of IQTools because it allows you to modify the code and experiment with the sequence and its settings.
- **4-Channel Synchronization (M8190A only)**: This example demonstrates how two M8190A modules can be synchronized. It works with or without the M8192A synchronization module.
- **16-Channel Synchronization (M8190A only)**: This example shows how to combine AXIe chassis, M8192A sync modules, and M8190A AWGs to a synchronous 16-channel system.

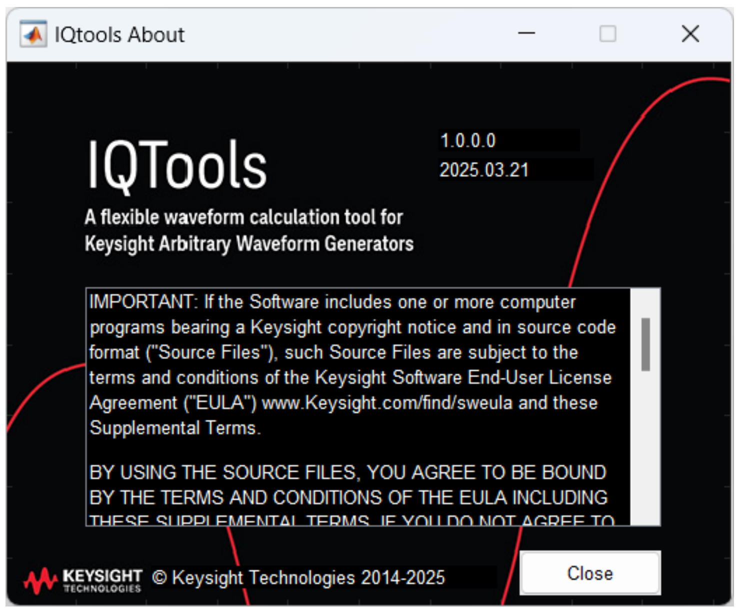
- **Digital Up-conversion Demo (M8190A only)**: This example demonstrates the digital up-conversion capability of the M8190A. It is best viewed with a spectrum analyzer connected to the M8190A output. If the spectrum analyzer is set up in IQTools configuration window, the example script will configure the spectrum analyzer as well as the AWG.
- **Band limited noise (M8190A only)**: This is an example application that shows how to generate band limited noise using the M8190A. The “randomness” of the noise signal is improved by using the M8190A sequencer.
- **Pulse Demo (M8195A only)**: This is a “pushbutton demo” of the M8195A capabilities to generate four synchronized pulse signals. It is best viewed with the four channels of an oscilloscope connected to the four channels of the AWG.
- **Angle of Arrival demo (M8195A only)**: This is a demonstration of the M8195A’s sequencing and capabilities. It is best viewed with the four channels of an oscilloscope connected to the four channels of the AWG.
- **FIR filter control (M8195A only)**: This is an example on how to control the FIR filters inside the M8195A.
- **Tone with calibrated amplitude (M8195A only)**: This example shows how tone power levels on the M8195A can be calibrated over frequency.
- **FMCW Radar Assistant (M8195A only)**: This is part of the E8742A FMCW Radar application. Refer to the *E8742A documentation* for further details.
- **ENOB measurement**: This example application performs an ENOB measurement of the AWG using a DCA or a spectrum analyzer.
- **KEYSIGHT Logo (M8190A and M8195A only)**: This “fun” application generates a specific signal, which resembles the Keysight logo when viewed on an oscilloscope. To view this output, you have to connect AWG channel 1 to scope channel 1 and AWG sample marker (in case of M8190A) or AWG channel 2 (in case of M8195A) to the scope’s AUX Trigger input.
- **Image to Spectrogram**: This “fun” application takes a .jpg file and generates a specific signal. When this signal is viewed on a spectrum analyzer in spectrogram mode, it resembles the original JPG picture.
- **Radar to Sound**: This is another “fun” application, which generates audible sounds in conjunction with the VSA software.


IQTools
File
IQ & RF
Digital
Other
Help
Configuration
Instrument Cor
IQ & RF
Multi-Tone &
Digital Modu
Radar Pulses
Frequency s
OFDM
Digital
Serial data generation
Sequencer Demo (M8190A only)
4-Channel Synchronization (M8190A only)
16-Channel Synchronization (M8190A only)
Digital Up-conversion Demo (M8190A only)
Band limited noise (M8190A only)
Pulse Demo (M8195A only)
Angle of Arrival demo (M8195A only)
FIR filter control (M8195A only)
Tone with calibrated amplitude (M8195A only)
FMCW Radar Assistant (M8195A only)
ENOB measurement
KEYSIGHT Logo
Image to Spectrogram
Radar to Sound

Help Menu

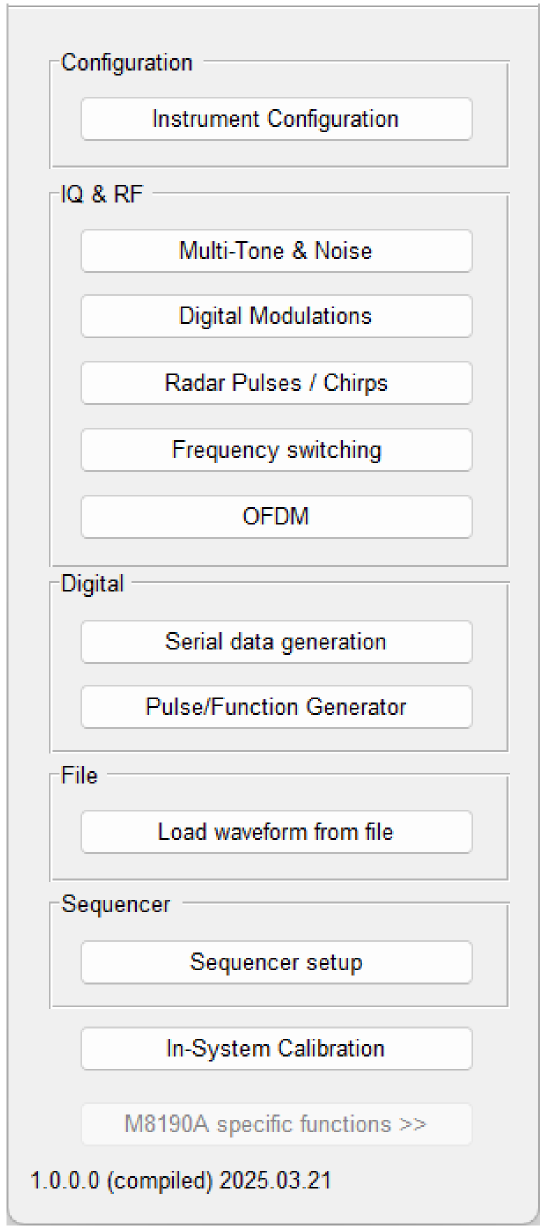
This menu opens the **IQTools About** window which provides information about the IQTools version, release date, and end-user license agreement.





Button Panel

The button panel refers to the area of the IQTools user interface that allows you to launch different applications. The buttons on the main window have the same functionality as the corresponding menu entries described above. Only the commonly used functions are available as buttons on the button panel. All other functions must be called through the menu.


Configuration
Instrument Configuration
IQ & RF
Multi-Tone & Noise
Digital Modulations
Radar Pulses / Chirps
Frequency switching
OFDM
Digital
Serial data generation
Pulse/Function Generator
File
Load waveform from file
Sequencer
Sequencer setup
In-System Calibration
M8190A specific functions >>
1.0.0.0 (compiled) 2025.03.21

The following buttons are available:

- **Configuration**
  - **Instrument Configuration**: Enables you to select the AWG or signal generator model from the list available and connect IQTools software with the AWG, signal generator, oscilloscope, and so on, by providing their visa addresses. For more information, see "Instrument Configuration" on page 64.
- **IQ & RF**
  - **Multi-Tone & Noise**: Allows you to create signals made up of one or more tones, either equally or arbitrarily spaced. It also allows for the definition of a frequency interval without tones (or notch) for NPR (Noise Power Ratio) testing. Amplitudes and phases of the individual tones can be corrected through correction factor files defined by the user. The Multi-Tone tab allows you to generate both RF and baseband (I/Q) signals. For more information, see "Multi-Tone and Noise" on page 71.
  - **Digital Modulations**: Allows you to create digitally modulated signals, both at IF/RF and baseband (I/Q). It directly supports a large variety of signal-carrier modulation schemes. For more information, see "Digital Modulations" on page 77.
  - **Dual-polarization Digital Modulations**: Allows you to create dual polarization digitally modulated signals. For more information, see "Dual Polarization Digital Modulation" on page 83.
  - **Radar Pulses/Chirps**: Use this option to create a variety of pulsed signals. It allows you to generate both RF and Baseband (I/Q) signals. It directly supports a variety of pulse train arrangements and intra-modulation schemes. For more information, see "Radar Pulses/Chirps" on page 89.
  - **Frequency Switching**: Allows frequency switching between two or more frequencies with adjustable tone duration and phase continuous switching. For more information, see "Frequency Switching" on page 96.
  - **OFDM**: This option enables signal generation with orthogonal frequency division multiplex modulation schemes. For more information, see "OFDM" on page 97.

- **Digital**
  - **Serial data generation**: Allows you to create single lane and multi lane bi-level and multi-level high-speed digital serial signals and clocks. It allows you to generate both data and clock signals. It directly supports a large variety of channel coding and modulation schemes. For more information, see "Serial Data Generation" on page 98.
  - **Pulse/Function Generator**: Allows you to generate any type of pulse by defining the pulse parameters in this section. For more information, see "Pulse/Function Generator" on page 111.
- **File**
  - **Load waveform from file**: Allows you to feed external data to IQ Tools by loading a waveform file. For more information, see "Load Waveform from File" on page 114.
- **Sequencer**
  - **Sequencer setup**: This is specific to M8190A, M8195A and M8198A. All the AWGs that support sequencing allow you to play sequential data by entering the required fields and defining the sequence as per your requirement. For more information, see "Sequencer Setup" on page 120.
- **In-System Calibration**: Allows you to calibrate the instrument for accurate waveform generation/acquisition by performing various analyses on the instrument, plotting various graphs, and extracting the data to display calibrated data. For more information, see "In-System Calibration" on page 131.
- **M81xxA specific function**: The functionality of this button changes depending on the selected AWG model. If you are using M8190A, M8195A or M8198A, you will find some model specific functions, such as CATV Signal Generation, Sequencer Demo (Time Domain), Radar Demo (Moving Target), 4-Channel Synchronization, and so on. For more information, see "M8190A Specific Functions" on page 152.

3 Working with IQTools

Concepts of the IQTools GUI

Small utility windows save screen real estate

The concept of the IQTools graphical user interface is to have separate, small windows (sometimes referred to as “waveform generation utilities”) for each type of waveform instead of a single, big application window that occupies the whole screen. Each utility window is optimized to occupy as little as possible screen real-estate and shows only the relevant parameters for the selected type of waveform.

This concept has been chosen because often times the signal analysis applications (for example, scope, spectrum analyzer, VSA, and so on) require a large portion of the screen.

Generate an AWG signal with as few as possible mouse clicks

A goal of the IQTools GUI design was to get the AWG to output signals with as few as possible mouse clicks: Whenever an IQTools waveform generation utility window is opened, the fields are populated with a set of meaningful parameters for the selected AWG. So, a simple “sanity check” is to open a waveform generation utility window and click on the “Download” button without changing any of the parameters. This should generate a signal on the AWG without throwing an error message. (Or, if you don’t have hardware attached or configured yet, click on “Visualize in MATLAB” to get a graphical representation of the waveform).

Concept of a “current setting”

Another concept in IQTools is to have a “current setting”, which is implicitly saved – and maintained across IQTools sessions. For example:

- In the configuration window, you can set up the VISA addresses of the AWG, scope, spectrum analyzer and so on. Once you click OK, this setting is used as the “current” setting. There is no need to save this setting to a file (although it is possible to do so, for example if you need to switch between two different hardware configurations).
- If you capture the frequency/phase response of the AWG using the “In-system calibration”, it is saved as the “current correction”. There is no need to save it to a file (but again, it is possible to do so).

Waveforms are generally calculated with complex values

Most of the waveform calculation routines work with complex-valued, that is, I/Q waveforms (hence the name IQTools). For real-valued signals (for example, a digital modulation signals that have been shifted to a carrier frequency), you can simply ignore the imaginary part.

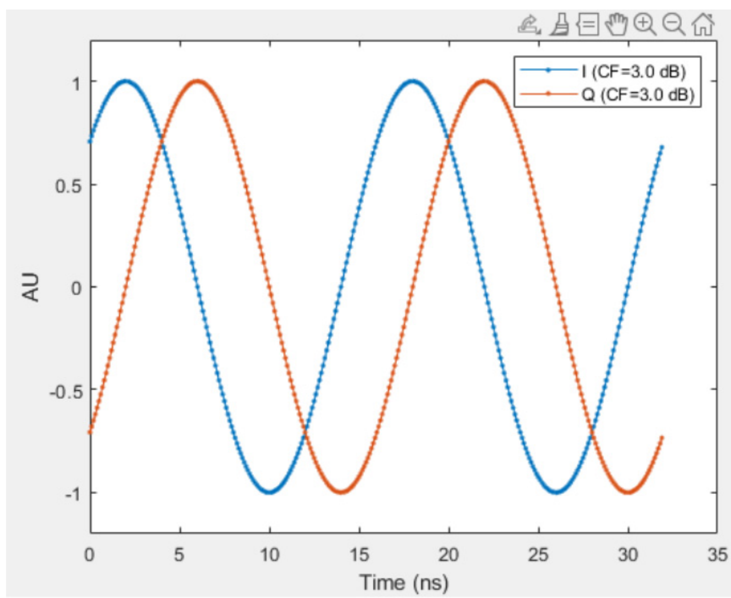
Common Features of Most Waveform Generation Utilities

While the set of parameter fields is specific to the individual waveform generation utility, there are some common features that appear on all waveform generation windows:

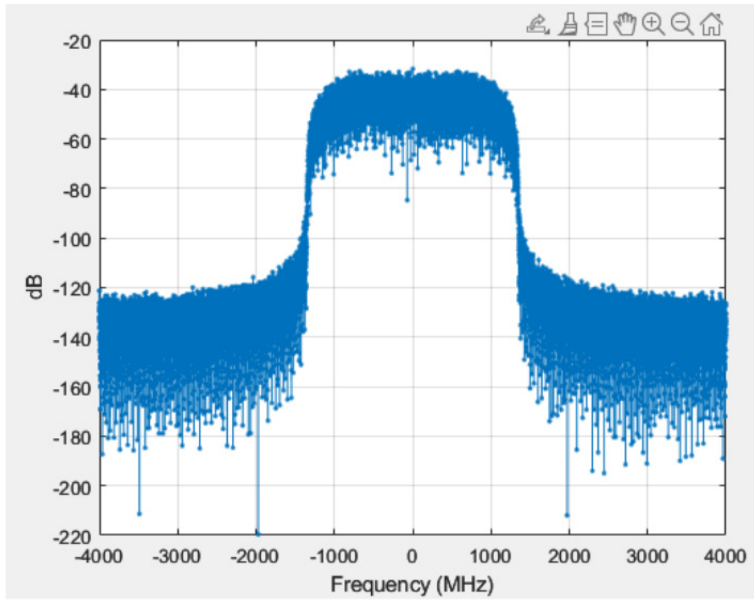
Visualize in MATLAB

The image below shows the calculated plots. Depending on the type of waveform, you shall see:

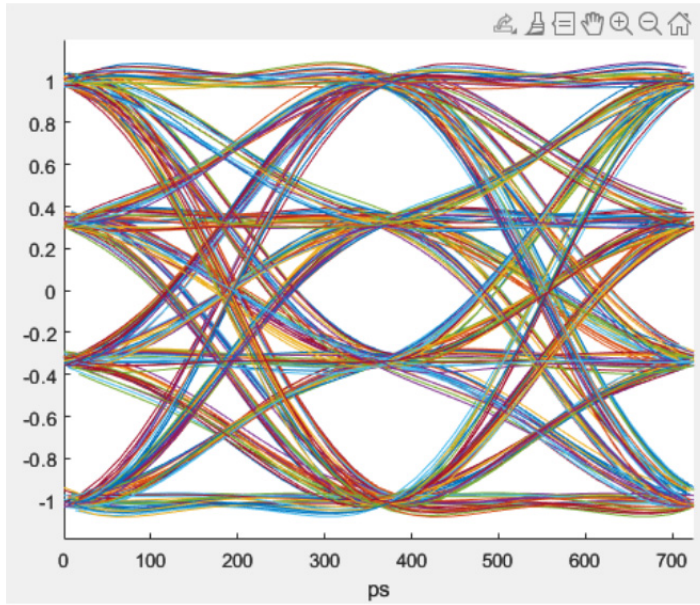
- **a time-domain plot**, showing the sample values over time. The individual sample values are shown as small dots that are connected by straight lines. Note that the actual signal coming out of the AWG will typically 'not' look exactly like that shown in the image below. This is due to the finite bandwidth of the AWG. At lower frequencies, the AWG does not linearly interpolate between the sample points (which is what the straight line might suggest). The Y-axis is labeled with AU (amplitude units). It corresponds to the internal waveform representation as floating point numbers in the range [-1 to +1]. If the waveform is complex valued, the real (I) and imaginary (Q) parts are shown in the same plot using blue and red colors. In the legend, the Crest Factor (CF) of the waveform is shown.



- **a spectrum plot**, which is basically an FFT of the calculated waveform. Note, that the Y-axis of the plot is in dB, not dBm. Hence, you cannot derive absolute power levels from this plot. Also note, that the noise level in these plots is due to the floating point accuracy of MATLAB. In the real signal, the noise from the AWG hardware will dominate.

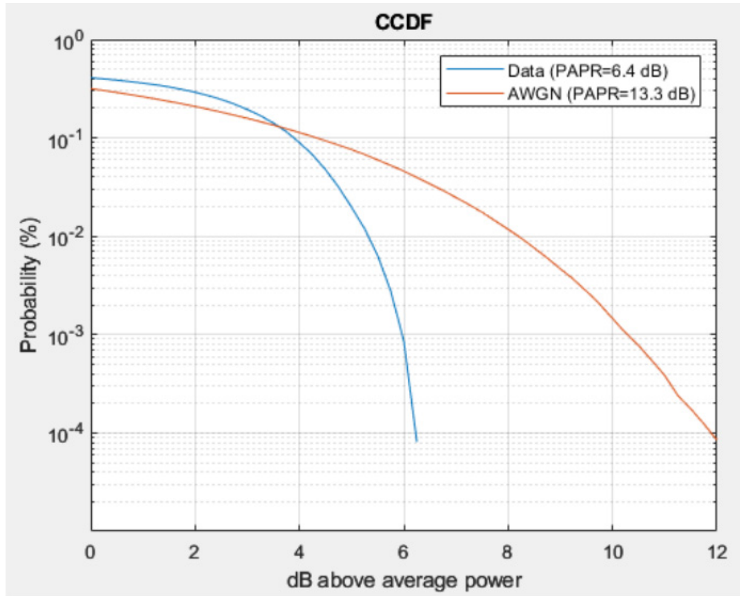


- **an eye diagram plot** (for serial data waveforms only). It shows all symbols in the waveform overlaid on top of each other. Unlike the time-domain plot, the waveform is not shown as individual samples with straight lines between them. Instead, the waveform is interpolated to show a smooth transition between the sample points which represents the behavior of the AWG output in a realistic manner, without taking its specific bandwidth limitations into account.



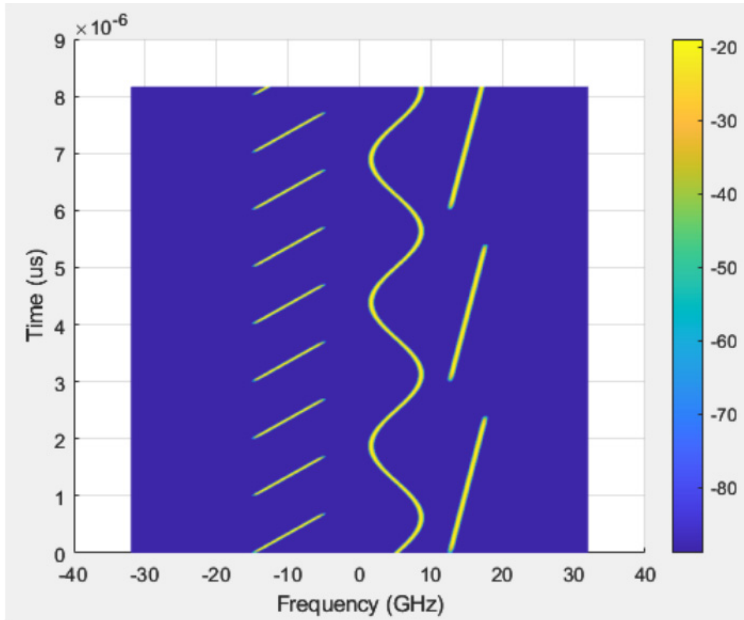
- **a CCDF plot** (multi-tone, noise and digital modulation only). The CCDF (Complementary Cumulative Distribution Function) plot shows the probability of the signal being at or above a certain power level. The power level is expressed in dB relative to the average power.

  In the CCDF plot, the blue curve is the current signal, the red curve shows the CCDF of a Gaussian noise signal with the same number of samples in comparison.



- **a spectrogram plot** (radar pulses and chirps, AM and FM modulation)

  It visualizes the magnitude (X-axis) over time (Y-axis). The intensity of the signal is represented through a color mapping. Just like the spectrum plot, the spectrogram plot shows only relative (dB), not absolute (dBm) power levels.



Note, that there is no hardware required to use the “Visualize in MATLAB” button.

Download button

This button loads the waveform into the AWG and starts playback. In order for the download functionality to work properly, the model, mode and VISA address of the AWG have to be set up in the “Instrument Configuration” window (see “Instrument Configuration” on page 64).

Channel Mapping

Most windows have a “Download To...” or “Channel Mapping” field. This setting is used to map the generated waveform to physical AWG channels. If the AWG operates in direct mode, one check box per column can be selected. If the AWG operates in digital up-conversion mode, a complex waveform (that is, both I and Q) are downloaded to the same AWG channel.

It is possible to load the same waveform into multiple AWG channels by selecting more than one check box per row. Note that the frequency/phase response correction is still applied to the physical AWG channel.

Sample Rate

Most windows have a “Sample Rate” field. This defines the sample rate for the AWG. IQTools is aware of the valid sample rate range for the selected AWG model and mode. If the sample rate value is outside of this range, the background of this field turns red. In some cases, there is an “Auto” check box next to the sample rate. Selecting this check box grays out the Sample Rate field and the IQTools algorithm will pick a sample rate value based on the remaining parameters.

Segment Number

Most windows have a “Segment Number” field. For AWGs that support multiple waveform segments and a sequencer, you can select the waveform segment into which the waveform will be downloaded. If the “Segment Number” field is grayed out, the configured AWG model does not support sequencing and multiple segments.

Apply Correction/Show Corrections

Most windows have an “Apply correction” check box and a “Show Corrections” button. If the Apply correction check box is selected, both the complex as well as per-channel corrections are applied to the waveform before it is visualized or downloaded into the AWG hardware. The “Show Corrections” button launches the correction management window, which is explained in detail in “Correction Management” on page 146.

Visualize in VSA

Some windows have a Visualize in VSA button. Clicking this button starts the VSA software, sets it up with the appropriate parameters to show the calculated waveform in VSA. To use this function, no hardware connection is required.

Save Waveform...

In the **File** menu of most utilities, you can find a **Save Waveform...** option. This can be used to save the calculated waveform in several file types: CSV, MATLAB, I/Q binary, and so on. Those files can be used for external post-processing/analysis or they can be loaded into an AWG at a later point in time through the “Load from file” utility. Note that it is not required to save a waveform to a file before it is downloaded. Saving just the parameter settings and re-creating the waveform on-the-fly offers more flexibility.

Load Setting/Save Setting

In the **File** menu of most utilities, you can find a **Load Settings...** and Save Settings... option. This function can be used to load/save the parameters of this window into a file.

Entering Numerical Values

All of the numeric input fields in IQTools will accept numbers in fixed point or engineering notation. Numbers without units are interpreted as seconds or Hertz, depending on the context. Optionally, unit indicators (s, sec, Hz), including the usual prefixes can be used in the edit fields for scalar values and simple lists of values, but not for MATLAB expressions. For example,

- 12e9 or 12g - to indicate 12 GHz
- 6.2e-9 or 6.2n or 6.2 ns - to indicate 6.2 Nanoseconds
- 3e6 or 3M or 3 MHz - is interpreted as 3 MHz. Note, that the “M” has to be a capital “M” to distinguish it from the “m” (milli, 1e-3) prefix
- 2e-12 or 2ps - will be interpreted as 2e-12 (2 pico)

Some fields which are marked with (*) in the graphical user interface also accept lists of values (that is, MATLAB vectors). Lists can simply be specified by multiple values separated by spaces or commas. For example,

- 500k, 1M, 2M - to indicate a list of frequency values (500 kHz, 1 MHz and 2 MHz. Since it is a simple list, unit prefixes are allowed.
- 1e6 * [0.5 1 2] - same as above. Note that “1e6” must be used in this case. 1 M is not allowed because it is a MATLAB expression.
- 1u, 3.5u, 10u - to indicate 1 us, 3.5 us and 10 us. Since it is a simple list, unit prefixes are allowed

In addition to numbers and lists of numbers, the fields also accept formulas using MATLAB syntax. For example,

- 1e9-5e3 - to indicate 5 kHz less than 1 GHz (or 999.5 MHz)
- 100e6:10e6:200e6 - a list of values from 100 MHz to 200 MHz in steps of 10 MHz
- linspace(100e6,200e6,11) - different way of describing the same as above: 11 values evenly spaced between 100 and 200 MHz
- 100e6 * rand(1,10) - 10 random frequencies between 0 and 100 MHz (“rand” is a built-in MATLAB function)

In addition to that, all of the input fields will evaluate variable names that are defined in the MATLAB workspace and formulas using MATLAB workspace variables. For example,

- Fs - can be used, for example, in the sample rate field if the variable “Fs” is defined in the MATLAB workspace
- 1e6*f - can be used to specify a list of frequencies assuming that “f” contains the list of frequencies in MHz
- fc-5e3 - center frequency minus 5 kHz – assuming that the variable “fc” is defined in the MATLAB workspace

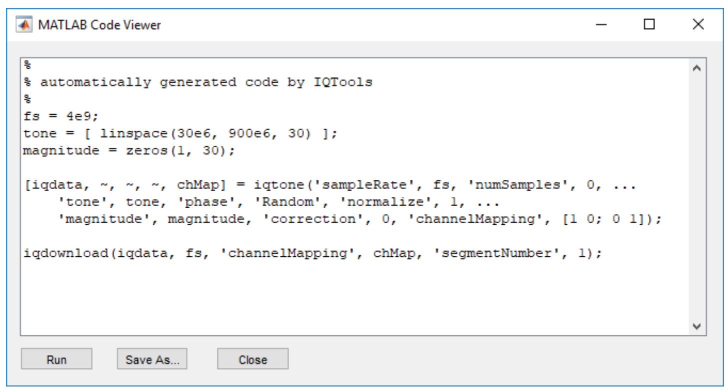
Working Directly in MATLAB without the IQTools GUI

Instead of using the graphical user interfaces of the scripts, IQTools functions can also be called directly from within your MATLAB script to generate, display and download waveforms. This can be useful to generate more complex waveforms, such as those, which may consist of multiple signals added together or downloading a waveform that has been calculated in a custom routine.

The “Generate MATLAB code” function that exists in most of the IQTools waveform utilities is a good starting point for a MATLAB script that calculates and downloads a waveform:

- Go to **File** > **Generate MATLAB code** on any of the IQTools utilities.

  The **MATLAB Code Viewer** screen appears.



The input parameters of iqtone, iqmod, iqpulse, iqfsk, iqpulsegen and iserial are specified as parameter/value pairs. All parameters have default values if they are not specified. You can look at the headers of the individual scripts for available arguments and ranges.

The scripts that can be used for your own programs are:

- iqtone - single and multitone
- iqnoise - generate noise signal
- iqmod - digital modulations
- iqpulse - pulsed RF signals incl. modulation on pulse (chirps, FMCW, Barker, and so on)
- iqfsk - frequency hopping signals

- iqpulsegen - function generator (pulses, ramps, and so on)
- iserial - serial data signals (NRZ, PAMx)

Waveform Display

The **iqplot** scripts can be used to display waveforms in both time and frequency domains. You can use iqplot to display real and I/Q data that has been generated by any of the waveform generation functions listed above or by your own MATLAB functions. Iqplot is called as follows:

```
iqplot(data, fs [, options])
```

The first parameter (data) is the desired signal represented as a vector of real or complex values in the range [-1...+1] and the second parameter (fs) is the sampling rate in Hz.

Optionally, you can specify further parameters that will influence how signals are being displayed:

- 'figure', N – the figure number to be used for plots (default is 1)
- 'no_timedomain' – the time domain plot will not be shown
- 'nospectrum' – the spectrum plot will not be shown
- 'smallspectrum' – only the “interesting” part of the spectrum will be shown (that is, the frequency range where the signal is above -90 dBm)
- 'spectrogram' – shows a spectrogram of the signal
- 'constellation' – shows the waveform as a constellation diagram
- 'no_CCDF' – no CCDF plot will be shown

Waveform Download

The **iqdownload** script is used to download waveforms into the AWG and start waveform generation. You can call iqdownload with a vector or array of real or complex data that has been generated by one of the waveform generation functions listed above or your own MATLAB functions. Iqdownload is called is follows:

```
iqdownload(data, fs [, option, value])
```

The first parameter (data) is the desired signal represented as a vector or array of real or complex values in the range [-1...+1] and the second parameter (fs) is the sampling rate in Hz.

By default, the real part of the data vector will be downloaded to channel 1 of the AWG and the imaginary part will be downloaded to channel 2. If you want the mapping to be different, see “‘channelMapping’” on page 60.

Do not call the download routines for individual AWGs directly, since the interface might change over time. (such as, iqdownload_M8190A, iqdownload_81180A, and so on.)

iqdownload accepts additional parameters as name/value pairs. These are explained in the header section of iqdownload.m. Some of them are explained in the following sections:

'channelMapping'

Using the 'channelMapping' (or 'chMap') parameter with iqdownload allows you to specify to which channel(s) the real and imaginary parts of a given column of your data will be downloaded. The channelmapping parameter must be a 2-dimensional array with 2*N columns, where N is the number of columns in your data array and M rows (representing the M channels of the AWG or synchronized AWG system). The "channelmapping" array consists of 2*N columns because you can define the real and imaginary parts of your data vectors to be downloaded into different channels. The individual values of the array must be either 0 or 1, to indicate whether the corresponding signal will be downloaded to a channel. Consider a waveform array with two columns and the following channel mapping array:

-	data column 1	data column 1	data column 2	data column 2	...
-	real part	imaginary part	real part	imaginary part	...
channel 1	1	0	0	0	
channel 2	0	1	0	0	
channel 3	0	0	1	0	
channel 4	0	0	1	0	
...					

(written in MATLAB terminology: [1 0 0 0; 0 1 0 0; 0 0 1 0; 0 0 1 0])

In this case, the real part of the data column 1 will be loaded to AWG channel 1, the imaginary part of column 1 to channel 2. The real part of data column 2 will be loaded into AWG channel 3 and 4 while the imaginary part of data column 2 is ignored.

In case of AWG setups that consist of multiple, synchronized modules, the channel numbers are counted continuously throughout the multi-module system; such as, in a setup with two M8195A modules and an M8197A sync module, the channels of the first M8195A are referred to as channels 1...4, the channels of the second M8195A are referred to as channels 5...8.

Note, that for DUC mode (if supported by the AWG), both real and imaginary part must be downloaded to a channel. A valid 'channelMapping' Parameter would for example be: [1 1; 0 0], which means real and imaginary part are loaded into channel 1 of the AWG.

'marker'

The 'marker' parameter allows you to specify the signal that will be generated out of the sample marker and sync marker signals. In general, the 'marker' parameter must be a vector with the same number of elements as there are samples in the waveform.

The values of the marker array are treated as integers where the lower four bits represent the four markers (sample1, sync1, sample2, sync2).

- A zero means that all markers are 0.
- A value of 15 (=binary 1111) means that all four markers are on.
- A value of 5 (=binary 0101) means that the two sample markers are on at the corresponding sample time).

Note that the markers in the M8190A cannot be arbitrarily turned on and off at any sample location, but will have a certain minimum pulse width (See datasheet for more details). An example MATLAB code that uses markers is shown here:

```
%

% automatically generated code from IQTools

%

fs = 1.2e+10;

iqdata = iqmod('sampleRate', fs, 'numSymbols', 3000,
...

    'modType', 'QAM16', 'oversampling', 12, ...

    'filterType', 'Square Root Raised Cosine',
'filterNsym', 80, ...

'filterBeta', 0.35, 'carrierOffset', 1e+09, ...
```

```
    'magnitude', [0], 'quadErr', 0, 'correction', 0);

% generate a marker vector with the same number of
elements as

% the analog waveform with a pulse on the sample
markers every

% 192 samples and a sync marker pulse at the beginning
of the waveform

n = length(iqdata);

pw = 192;

cnt = floor(n / pw);

marker = repmat([5 * ones(1, pw/2) zeros(1, pw/2)], 1,
cnt);

marker(1) = 10;


iqdownload(iqdata, fs, 'channelMapping', '[1 0; 0 1]',
...

    'segmentNumber', 1, 'marker', marker);
```

Generating RF/IF Waveforms

All of the IQTools waveform generation utilities (except iserial) will generate I/Q baseband or direct IF/RF waveforms. In case of baseband I/Q waveforms, the “I” signal and “Q” signals are intended to be connected to an I/Q modulator, such as the wideband I/Q inputs of a Vector PSG (E8267D) or an optical modulator. They can also be connected directly to an oscilloscope for analysis of the I/Q waveform using the VSA software.

However, the tools can also be used to generate an IF/RF signal directly. In order to generate an RF signal, make sure that:

- For **Multi-Tone**, both start and stop frequency are positive.
- For **Digital Modulation signals**, the carrier offset is positive and larger than ½ of the bandwidth of the modulated signal.
- For **Radar chirps**, frequency offset is positive and larger than ½ the frequency span.
- For **Frequency Switching**, all frequencies in the list must be positive.

The simulated spectrum that is shown when the **Display** button is pressed should only show positive frequency components.

iserial is an exception and always generates a real-valued signal.

NOTE

Launch the IQTools application using the “Run as Administrator” right-click option to establish successful connection and operation with the M5300A RF AWG module. If you do not have administrative privileges on your machine, contact your local IT administrator.
For information about how to use load waveform files in the M5300A RF AWG modules, refer to the *M5300A PXIe RF AWG User Guide*.

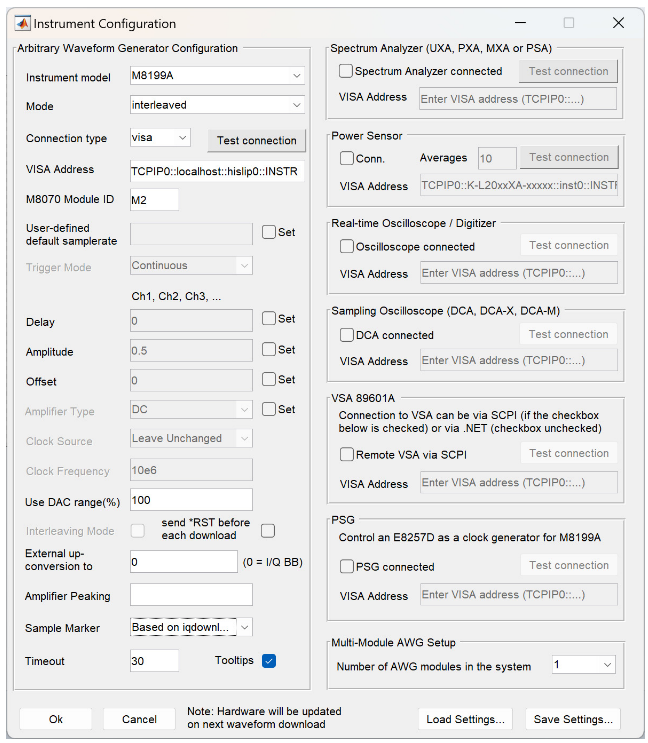
Instrument Configuration

The instrument configuration screen enables you to select the required instrument from the list available and connect IQ Tools software with the instrument by providing the visa addresses of the instrument. The instrument can be the generator, oscilloscope, or any other analyzer listed in IQ Tools.

To connect an instrument with IQTools, do the following:

1 Under the **Configuration** pane, click **Instrument Configuration**.

The **Instrument Configuration** screen appears.



2 Select an instrument from the **Instrument model** drop-down list. For some models, you must select which mode you want the AWG or signal generator to operate in.

3 Based on the instrument model selected, select an appropriate mode from the **Mode** drop-down list.

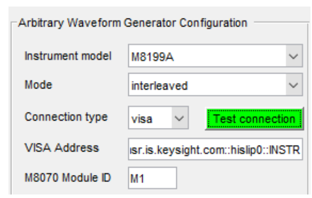
4 Select an option from the **Connection type** drop-down list. The following options are available:

- tcpip - Use 'tcpip' for direct socket connections.
- visa - Use 'visa' for connections through VISA library

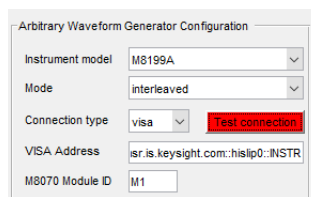
5 Enter the **VISA Address** as given as in the Keysight Connection Expert.

6 Click **Test Connection** to check the connection.

In case of a successful connection, the **Test Connection** button is green in color.



In case of an unsuccessful connection, the **Test Connection** button is red in color.



7 Depending on the selected AWG model and mode, you can configure the other available parameters available on the **Instrument Configuration** screen.

Most of the additional parameters have a "Set" check box next to them. If the "Set" check box is not selected, this parameter is not touched by IQTools, that is, it remains at its default value or whatever value it was previously configured, such as, in the instrument SFP. If the "Set" check box is selected, the corresponding parameter is overwritten by the value specified in the IQTools configuration window.

Most of the parameter fields accept either a single value, which is used for all AWG channels or a list of multiple values separated by space or comma. In this case, the first value is used for AWG channel 1, the second for channel 2 and so on. Note that changes to any of the fields are only reflected in the hardware after you download a new waveform from any of the waveform generation windows.

The following fields are available:

- **User-defined default sample rate**: When opening any of the IQTools waveform generation windows, the “Sample Rate” field is populated with the factory default sample rate for this AWG. If you enter a value into the “User-defined default sample rate”, that value is used as a default value when opening waveform generation windows. You may still overwrite the sample rate with a different value if desired.
- **Trigger Mode**: For AWGs that support triggering, you can select between “Continuous”, “Triggered”, “Gated” or “Leave unchanged”.
- **Delay**: For AWGs that support this function, you can set the (analog) delay for each channel in this field.
- **Amplitude**: For AWGs that support adjustable amplitude, you can set the amplitude for all channels or for each channel individually.
- **Offset**: For AWGs that support adjustable offset, you can set the amplitude for all channels or for each channel individually.
- **Amplifier Type**: For AWGs that support multiple output paths, you can select the desired amplifier path.
- **Clock Source**: For AWGs that support different clock/reference clock sources, you can select the desired clock source/reference clock source.
- **Clock Frequency**: When selecting an external clock source or external reference clock source, you can specify the clock/reference clock frequency.
- **Use DAC range**: Use this field to specify what percentage of the DAC full-scale range will be used. Usually, this field should be left at 100% (that is, use the full range).
- **Interleaving Mode**: Select this check box if you externally interleave two AWG channels. This function is only supported and tested with the M8190A. For the M8199A, DO NOT select this check box, but use the mode selection to configure interleaved mode.

- **Send *RST before each download**: If you want to make sure that the AWG is in a well-defined state before downloading a waveform, you can select this check- box. It is strongly recommended to keep this check box cleared during normal operation because some tools perform multiple download operations in sequence. If this check box is selected, these won't work because the AWG is reset every time.
- **External up-conversion**: You can enter the LO-frequency of an external up-converter in this field. This will not change the behavior of the waveform calculation, but it will populate the "Fc" field. For example, in the Digital Modulation and Radar Pulse utility with the appropriate value.
- **Amplifier Peaking**: Some AWG models support variable amplifier peaking. Positive peaking values increase the bandwidth, negative values decrease the bandwidth relative to the factory setting. You can specify a list of values separated by space or comma to define a different peaking value per channel. Note that changing amplifier peaking to a non-zero value is considered as over-programming and is outside of the specification.
- **Sample Marker**: In this field, you can define a certain clock pattern to be generated on the M8198A/M8199A/M8199B sample marker output – independent of the waveform that is being downloaded. This is useful if you use the M8198A/M8199A/M8199B sample marker as a clock source for the DCA PTB/Front Panel Trigger.
- **Timeout**: This field specifies the connection timeout with the AWG in seconds. Usually, 10 – 30 seconds are sufficient, but for instance, with remote connections, longer timeout value may be necessary.

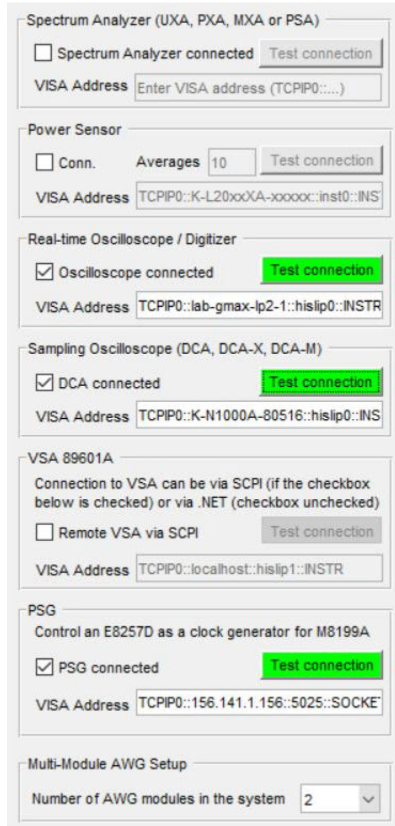
8 You can select the **Tooltips enabled** check box to enable the tooltips throughout IQTools.

   The tooltip is a small pop-up window that concisely describes the object being pointed to, such as descriptions of the parameters.

9 Click **OK** when the connection is done successfully.

   The instrument is now connected.

Similarly, you can connect the other instruments like spectrum analyzer, power sensor, real-time oscilloscope/digitizer, sampling oscilloscope, VSA and so on, using the **Instrument Connection** screen.



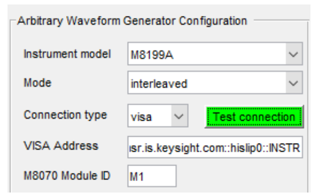
1. Select the respective check box for the instrument.
2. Enter the **VISA Address**.
3. Click **Test Connection** to check the connection.

   The **Test Connection** button shows green color when the connection is successful.
4. Click **OK**.

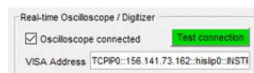
   The instrument is now connected.

Multi-Module AWG Setup

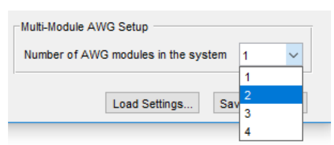
1 Select an instrument from the **Instrument model** drop-down list and set-up a valid connection.



2 Connect the real-time oscilloscope/digitizer on the **Instrument Connection** screen.



3 After you connect the instrument and the real-time oscilloscope/digitizer, select the number of AWG modules in the system from the drop-down list.

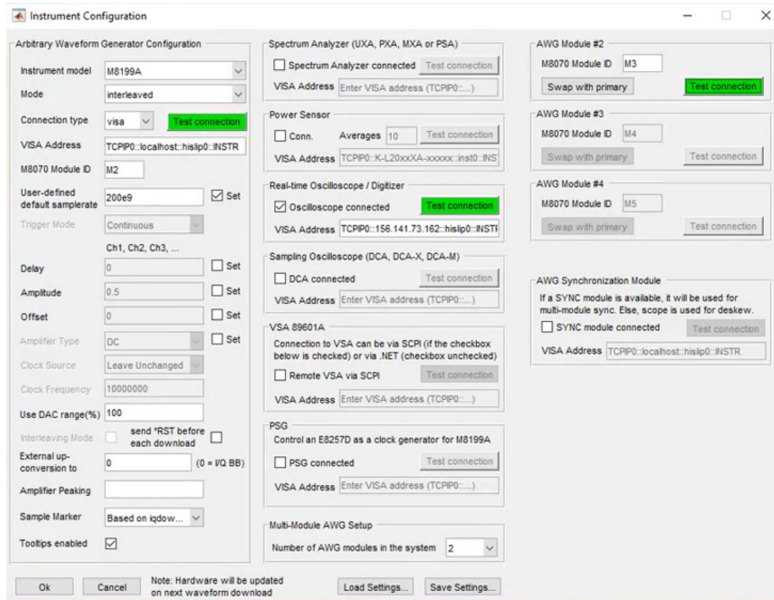


4 Connect the selected modules and then click **OK**.

The multi-module setup is now ready.

Within IQTools, channel numbers are assigned consecutively, throughout the multi-module setup, starting from 1.

Example: in a setup that consists of two M8199B modules, the channels of the first M8199B module are referred to as channels 1 and 2, while the channels of the second M8199B modules are referred to as channels 3 and 4.


Instrument Configuration
Arbitrary Waveform Generator Configuration
Instrument model
M8199A
Mode
interleaved
Connection type
visa
Test connection
VISA Address
TCPIP0::localhost::hislip0::INSTR
M8070 Module ID
M2
User-defined default samplerate
200e9
Set
Trigger Mode
Continuous
Ch1, Ch2, Ch3, ...
Delay
Amplitude
Offset
Amplifier Type
DC
Clock Source
Leave Unchanged
Clock Frequency
10000000
Use DAC range(%)
100
Interleaving Mode
send *RST before each download
External up-conversion to
(0 = I/Q BB)
Amplifier Peaking
Sample Marker
Based on iqdow...
Tooltips enabled
Spectrum Analyzer (UXA, PXA, MXA or PSA)
Spectrum Analyzer connected
Enter VISA address (TCPIP0::...)
Power Sensor
Conn.
Averages
10
TCPIP0::K-L20xxXA-xxxxx::inst0::INS
Real-time Oscilloscope / Digitizer
Oscilloscope connected
TCPIP0::156.141.73.162::hislip0::INSTR
Sampling Oscilloscope (DCA, DCA-X, DCA-M)
DCA connected
VSA 89601A
Connection to VSA can be via SCPI (if the checkbox below is checked) or via .NET (checkbox unchecked)
Remote VSA via SCPI
PSG
Control an E8257D as a clock generator for M8199A
PSG connected
Multi-Module AWG Setup
Number of AWG modules in the system
2
AWG Module #2
M8070 Module ID
M3
Swap with primary
AWG Module #3
M4
AWG Module #4
M5
AWG Synchronization Module
If a SYNC module is available, it will be used for multi-module sync. Else, scope is used for deskew.
SYNC module connected
TCPIP0::localhost::hislip0::INSTR
Ok
Cancel
Note: Hardware will be updated on next waveform download
Load Settings...
Save Settings...

Multi-Tone and Noise

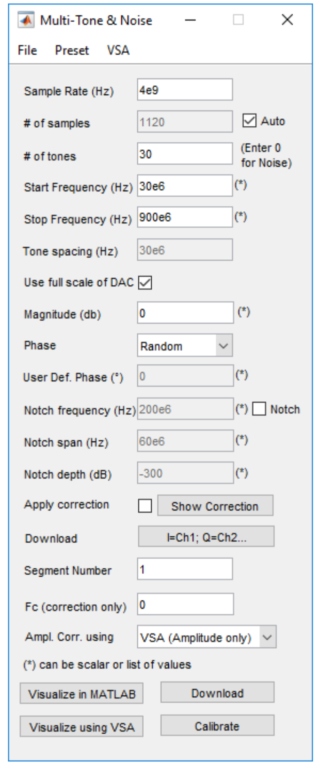
The **Multi-Tone and Noise** screen can be used to generate single tone, two tone, multi tone signal with equidistant or non-equidistant spacing as well as band-limited noise. “Notches” with variable width and depth can be included with multi-tone or noise signals.

The algorithm in the multi-tone utility always calculates complex-valued signals (I & Q). If you need a real-valued (non-I/Q) signal, simply choose your tone frequencies to be all positive (or all negative). In this case, you can simply ignore the “Q” row in the channel-mapping dialog.

If you use an AWG in conjunction with an I/Q modulator (such as, M8190A connected to a Vector PSG), your range of tones can be from – FS/2 to + FS/2. The I/Q modulator will move the frequency range to its carrier frequency.

- To use the multi-tone utility, click **Multi-Tone & Noise** under the **IQ & RF** pane.

  The following figure displays the **Multi-Tone & Noise** parameters:



In the following sections, some special cases are described that can be covered by the multi-tone utility.

Noise

To generate a noise waveform in a certain frequency band:

- Set the number of tones to zero.
- Choose a large number of samples (1000000 is a good starting point). The larger your number of samples, the more “random” is your noise. On the other hand, a larger number of samples takes longer to calculate and download the waveform.

The **Start** and **Stop** frequency fields can be used to limit the bandwidth of the noise. If you check the **Notch** check box, this feature can be used to generate a spectral “gap” in the noise signal with adjustable center (Notch Freq. field) width (Notch span) and depth (Notch depth). It is possible to specify multiple notches with different center frequencies, widths, and depths. By specifying adjacent notches, it is possible to generate a fully used-definable noise spectrum.

Equidistant and Non-Equidistant Tones

If you just select the number of tones and a start and stop frequency, the tones will be distributed equidistantly between the start and stop frequency. The algorithm will put a tone on the start and stop frequency itself. You should take this into account when selecting the number of tones. For example, if you want equidistant tones from 1 GHz to 2 GHz with 10 MHz spacing, you should choose 101 tones (not 100).

For non-equidistant tones, set the number of tones to “1” and enter the list of tone frequencies that you would like to generate in the **Stop frequency** field. You can use MATLAB expressions such as:

100e6 * [-3 -1 4 5 7.5 9]

This will generate tones at -300, -100, 400, 500, 750 and 900 MHz.

Tones with Different Amplitudes

You can use the **Notch** feature to generate tones at different relative amplitudes. You must only define a Notch for each tone (or range of tones) that you want at a different level.

Example:

1 Set the **Number of tones** to 1 and the **Stop Frequency** to: `100e6 * [-3 -1 4 5 7.5 9]` to generate tones at -300, -100, 400, 500, 750 and 900 MHz. Without anything else, they will all have the same relative amplitude.
2 Check the **Notch** check box, and enter `100e6 * [-3 -1 4 5 7.5 9]` in the **Notch frequency** field.
3 Change the **Notch span** to `1e6` (can be any number > 0 because you just want to hit that specific tone) and enter `-5 +5 -10 +10 0 0` in the **Notch depth** field. This will change the relative amplitude of the tones.

   This works both with individual tones as well as with tone ranges.
4 To see an example with tone ranges at different levels, click on **Preset** > **Multi-tone** with multiple Notches in the Multi-tone utility.
5 Click **Visualize in MATLAB** or **Download** to see the result in MATLAB resp. in hardware.

Generating a Different Multi-Tone Signal on Each AWG Channel

It is possible to open multiple multi-tone windows side-by-side by clicking the “Multi-tone and noise” button in the IQTools main window multiple times. This allows for example to generate a different multi-tone signal on each AWG channel without having to re-configure the parameters every time.

Multi-Tone Amplitude or Amplitude and Phase Correction

To improve flatness for multi-tone signals, the “iqtone” utility offers a number of amplitude (resp. amplitude/phase) correction functions.

Through the pull-down menu, you can choose between different algorithms and analysis instruments. Independent on which algorithm you choose, IQTools controls the AWG as well as the analysis instrument to perform the measurement. It is important to first configure the access to the analysis instrument (using the IQTools config window) prior to making the measurement. By nature, the spectrum analyzer and power sensor measurements correct for amplitude only. Using VSA, it is possible to measure amplitude and phase.

Each correction function has specific pros and cons as shown in the table below:

Value in popup-menu	Description
Spec.An (zero span)	Spectrum analyzer uses “zero span” mode. While the AWG generates the desired multi-tone signal, the SA center frequency is set to one tone at a time. **Pro**: reasonable measurement speed, resilience against small tone deviations (for example, if AWG and SA are not frequency locked) **Con**: SA’s frequency response behavior is different between zero-span mode and regular swept mode, which leads to inaccuracies
Spec.An (markers)	Spectrum analyzer is set to swept mode. AWG generates the desired multi-tone signal. An SA “Marker” is used to measure the magnitude of each tone. **Pro**: fastest measurement speed, consistent results (because SA is always in swept mode) **Con**: due to small frequency deviations between AWG and SA, markers sometimes don’t hit the peaks of tones
Spec.An (list sweep)	AWG generates the desired multi-tone signal. Spectrum analyzer uses “List Sweep” mode to measure all tone magnitudes at the same time. **Pro**: fastest measurement speed **Con**: same as zero span
Spec.An (single tone)	AWG is programmed to generate one tone at time instead of the whole multi-tone signal.

Value in popup-menu	Description
VSA (amplitude only)	AWG generates the desired multi-tone signal. The VSA software is configured to capture the whole bandwidth of the signal and uses a marker to measure the magnitude of each tone. **Pro**: works with SA or oscilloscope **Con**: same as "Marker" mode with spectrum analyzer
VSA (amplitude and phase)	AWG generates the desired multi-tone signal. The VSA software is configured to capture the whole bandwidth of the signal and measures magnitude and phase of each tone. **Pro**: can measure (and correct for) amplitude and phase **Con**: same as "Marker" mode with spectrum analyzer
Power sensor	AWG is programmed to generate one tone at time instead of the whole multi-tone signal. Power sensor is used to measure the power of each tone. **Pro**: power sensor yields more accurate magnitude results compared to a spectrum analyzer

Once the measurement has completed, the result is stored in IQTools and can be used to correct for amplitude/phase compensation in the waveform calculation routine.

This amplitude correction can later on also be used by the other waveform creation utilities to improve the flatness. The amplitude flatness correction can compensate non-flatness on the output of the AWG or perform correction even after I/Q up-conversion in a signal generator. In this case, the magnitude response of the signal generator is also compensated.

To perform an amplitude correction:

1 Use the **Configure Instrument Connection** tool to set up the communication parameters for the AWG, spectrum analyzer/oscilloscope/VSA.

2 Choose the desired measurement algorithm in the popup menu (see table above).

3 Set the Start/Stop/number of tones parameter to generate a multi-tone signal that spans the entire frequency range that need to be corrected later on. Do not configure any notches for the correction measurement.

   A good starting point for the M8190A is the Preset **101 tones, +/- 1 GHz, asymm**. (can be found in the menu bar) if you are generating I/Q data. For a direct IF/RF signal (not using I/Q), choose a range of frequencies that are all positive. A larger number of tones will improve the accuracy, but increase the measurement time. Around 100 – 200 tones produces reasonable results. For demo purposes, 30 or 40 tones are good enough. To get optimal results, the **Phase** should be set to **Random** and when using I/Q up-conversion, the tones should be selected such that the images don't fall on top of generated tones. (that is, don't use symmetrical intervals around 0 Hz).

4 Click **Download** once to see how the un-corrected waveform looks like. The multi-tone calibration works either with direct AWG output or with I/Q output going into the wideband modulation inputs of a PSG and measuring the PSG output. (Ensure that **Apply Correction** check box is clear).

5 Set the center frequency (“Fc – for calibration only”) in the Multi-Tone GUI to zero if you analyze the direct AWG output. If you have the I/Q signals connected to a PSG and correct the flatness of the final RF output, enter the Carrier frequency set in the PSG. If you are up-converting using a mixer, enter the difference between the LO and IF frequency (LO – IF) if you are looking at the mixer’s upper sideband. For the lower sideband, enter -1 * (LO + IF).

   Ensure that **Apply Correction** check-box is clear to start a new correction measurement.

6 Click the **Calibrate** button. Depending on the selected measurement mode, the spectrum analyzer or oscilloscope will perform a measurement. The result is used as the “current” correction. (The current correction result is stored in the file *C:\users\<username>\AppData\local\Keysight\iqtools\ampCorr.mat*. Usually, there is no need to access this file directly). The measured frequency response is also displayed as a MATLAB plot. This plot is just for your information and can be closed if it no longer needed.

   To further improve the calibration, you can run it again, taking the previous correction factors into account. Experience shows that after running the calibration about three times, no further improvement can be achieved. The red graph indicates the measured deviation for each tone from the average tone power. It should get close to the zero line as you perform repeated calibrations.

7 The saved correction factors can be used by the other utilities (iqmod, iqpulse, iqfsk) as well. Make sure to enable the **Apply Correction** in the respective utility.

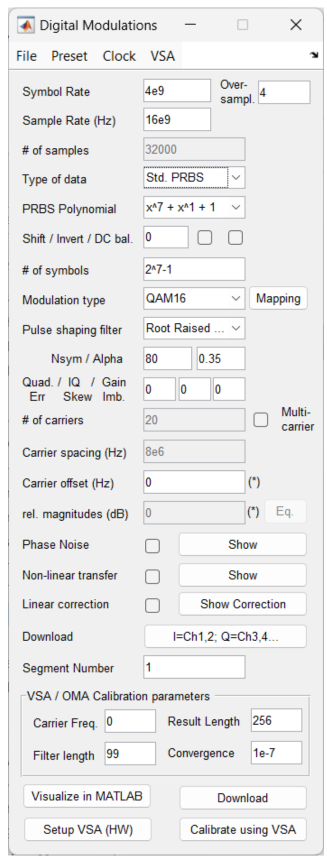
NOTE When performing the amplitude correction, ensure that the start and end frequencies span at least the same range of frequencies that you will use later on in other utilities.

Digital Modulations

This utility allows you to generate digitally modulated waveforms with selectable modulation schemes, pulse shaping filters and impairments – either as I/Q baseband or as RF waveforms. The utility supports integer and fractional oversampling, single or multi-carrier waveforms with user-definable relative magnitudes. It also supports amplitude and phase correction in conjunction with the VSA software.

- To use the Digital Modulation utility, click **Digital Modulation** under the **IQ & RF** pane of the IQTools main window (or select Digital modulation in the IQ & RF menu).

  This figure shows the **Digital Modulation** utility:



The following parameters are available:

- **Sample rate**: Can either be entered manually or it will be calculated when changing the oversampling rate.
- **Oversampling ratio**: Usually, this field does not need to be set directly, but if a certain oversampling ratio is desired, a value can be entered. In this case, the sample rate will be re-calculated to match with the symbol rate and oversampling ratio.
- **Symbol rate**: Enter the desired symbol rate (baud rate). If you change the symbol rate, the algorithm will calculate the oversampling ratio, which is shown as a fraction (unless you selected a symbol rate which happens to divide evenly into the sample rate). You should pay attention to the calculated oversampling ratio. If the numerator or denominator are large (> 100), the calculation time for the waveform will go up significantly. In this case, it might be better to use a different sample rate or overwrite the oversampling ratio if that is possible.
- **Number of Samples**: Is automatically calculated based on the number of symbols times the oversampling ratio. If this result does not match the AWG's granularity requirements, the waveform is repeated multiple times, which increases the number of samples.
- **Type of Data**: can be selected from:
  - **Random**: Uses the MATLAB "rand" function to generate a random set of symbols.
  - **Clock**: Toggles between the smallest and largest possible symbol value.
  - **Counter**: Iterates through all possible symbol values.
  - **Std. PRBS**: Generates a PRBS pattern with a standardized polynomial. The type of PRBS can be selected in the "PRBS polynomial" popup menu.
  - **Custom PRBS**: Generates a PRBS pattern with a user-defined polynomial. The polynomial can be entered in the "PRBS polynomial" edit field.
  - **User Defined Symbols**: Allows you to enter a sequence of symbols. If the modulation format has M symbols, the values in the "Data content" field are expected to be in the range from 0 to M-1. For details on the syntax, see "Data Content" below.
  - **User Defined Bits**: Allows the user to enter a sequence of bits. The bits are converted to symbols. For details on the syntax, see "Data Content" below.

   - **Symbols from file**: Allows the user to enter the name of a file with symbol values in it. The values in the file are expected to be numbers from 0 to M-1 (where M is the number of symbols in the selected modulation format).
   - **Bits from file**: Allows you to enter the name of a file with bit values in it. The values in the file are expected to be 0 or 1.
- **PRBS polynomial**: Select the desired PRBS polynomial (Std. PRBS) or enter the user-defined polynomial (Custom PRBS). The format for entering a polynomial is either:

  x^p + x^q + ... + 1

  or

  [p q ... 0]
- **Shift/Invert/DC-balanced**: "Shift" defines the number of symbols by which the pattern is (cyclically) shifted. This is particularly relevant for dual polarization digital modulation to avoid identical patterns in the X- and Y polarization.

  "Invert" inverts the bits of the PRBS pattern.

  "DC-balanced" inserts an additional zero at the longest stretch of zeros in the PRBS. This causes the pattern length to be 2^N instead of 2^N-1. It also generates a stretch of all zeros with length N. In a regular PRBS, the longest stretch of 1s is N, whereas the longest stretch of zeros is N-1.
- **Data Content**: If "Data Type" is set to "User defined symbols" or "User defined bits", this field can be used to specify a vector of symbols resp. bits. Data can be entered either as a list of values separated by comma or space or as a MATLAB expression that evaluates to a vector.

  Examples:

  0 1 0 1 1 1 0 0 1 0 1 1 0 1 is a sequence of bits (or NRZ symbols).

  3 2 0 1 2 2 3 0 0 1 1 2 is a sequence or PRM-4 symbols.

     - randi([0 3], 40, 1) is a MATLAB expression that evaluates to a random sequence of 40 PAM-4 symbols.
- **Filename**: If "Data Type" is set to "Symbols from File" or "Bits from File", this field is used to specify the filename that contains the symbols resp. bits. The values in the file are expected to be in the range 0 to M-1 for symbols or 0 and 1 for bits.
- **Number of symbols**: Defines the number of symbols that will be encoded in the waveform if Type of data is Random, Clock, Counter or PRBS 2^N-1. For other types of data, the number of symbols is derived from the file or the Data content field.

- **Modulation Type**: Select from modulation formats BPSK, QPSK, QAM-n, APSK-n. The “Mapping” button next to the modulation format popup menu can be used to visualize and modify the constellation diagram. See “Custom Modulation” on page 84.
- **Pulse shaping filter**: Select between Raised-cosine or Root-raised-cosine pulse shaping filter. Filter types Gaussian, Rectangular and None are not tested.
- **NSym/Alpha**: The NSym field defines the length of the pulse shape filter in number of symbols. The Alpha field defines the roll-off of the pulse shape filter (0 < alpha <= 1). The smaller the alpha value, the larger NSym should be chosen to avoid signal discontinuities. When the alpha value is changed, IQTools replaces the NSym value with the recommended value in case it is smaller.
- **Quad Err/IQ Skew/Gain Imb.**: These fields can be used to add distortions to the signal. **Quad Err** (Quadrature Error) is given in degrees. It modifies the nominal 90° angle between I and Q by a certain amount. **IQ Skew** introduces a skew between I and Q. The value is given in units of seconds. **Gain Imbalance** modifies the magnitude of the Q signal relative to the I signal. Gain imbalance is specified in dB.
- **Multi-Carrier**: This check box turns on multi-carrier mode. When is turned on, the “# of carriers” and “Carrier Spacing” fields become relevant. In multi-carrier mode, a number of carriers is generated simultaneously at different center frequencies.
- **# of carriers**: Specifies the number of carriers in multi-carrier mode.
- **Carrier Spacing**: Specifies the distance between center frequencies in multi-carrier mode.
- **Carrier Offset**: If the Carrier offset field is zero, an I/Q baseband signal is generated. If carrier offset is set to a non-zero value, the waveform will be shifted to the specified frequency. The carrier offset is specified in units of Hertz. Note that the actual carrier offset frequency may be rounded such that an integer number of carrier periods is used for the calculated waveform (that is, no phase discontinuities are being generated).

  When multi-carrier mode is turned on, the carrier offset defines the center frequency of the first (= lowest) carrier. The center frequencies of the carriers are given by (carrier_offset + N * carrier_spacing), where N = 0 ... number_of_carriers – 1.

- **Rel. Magnitudes**: In multi-carrier mode, this field is used to specify the magnitudes of the carriers relative to each other. Values are in dB and should be entered as a vector of values, separated by space or comma. Note, that the absolute magnitude of carriers will be reduced the more carriers are being generated simultaneously. If fewer values than number of carriers are specified, the list is repeated as often as necessary.
- **Phase Noise**: When the check box is selected, adds a certain amount of pseudo-random phase noise to the signal. The magnitude and characteristics of the phase noise can be entered by clicking the "Show" button.
- **Non-linear transfer**: When the check box is selected, a non-linear transfer function is applied to the signal. The shape of the non-linear transfer function per channel can be defined by clicking the "Show" button.
- **Linear correction**: When the check box is selected, the linear correction (result of the in-system calibration or VSA calibration) is applied to the signal.
- **VSA Calibration Parameters**: The fields in this pane are only relevant if the VSA software is used for magnitude/phase response correction.
    - **Fc for correction**: Without external up-conversion, the value should be identical to "Carrier offset". With external up-conversion, this value should be set to the center frequency that VSA should measure at. Typically, this is the frequency of the LO plus or minus the Carrier Offset.
    - **Result length**: This value is used to set the Result length in VSA.
    - **Filter length**: This value is used to set the equalizer filter length in VSA during a freq/phase response correction measurement.
    - **Convergence**: This value is used to set the equalizer convergence in VSA during a freq/phase response correction measurement.
- **Setup VSA (HW)**: This button configures the VSA software for digital demodulation with the parameters that are set in this utility (symbol rate, modulation format, filter type, filter beta, Carrier offset, Fc, result length). The button is particularly useful for a quick demo of the AWG waveform.
- **Calibrate using VSA**: This function is described in the paragraph "Amplitude and Phase Corrections for Digital Modulation Waveforms".

Amplitude and Phase Corrections for Digital Modulation Waveforms

When generating a digitally modulated signal with the **Digital Modulation** utility, you can improve the EVM by performing an amplitude and phase calibration in conjunction with the VSA software. The VSA software has to be installed on the same PC that runs the MATLAB scripts. The connection to the oscilloscope that captures the signal must be established before using the calibration function in the MATLAB script. The calibration routine uses the Equalizer that is built into the VSA software to determine the channel frequency response. After generating an (un-corrected) signal, the MATLAB script launches the VSA software, turns on the equalizer and uses the frequency response of the equalizer to calculate a pre-distorted waveform. Unlike the flatness correction using multi-tone, this method corrects magnitude and phase of the signal.

Follow these steps to generate a pre-distorted signal:

1. Set the desired parameters in the **Digital Modulation** tool and click download to generate a digitally modulated signal. Ensure that the **Apply Correction** check-box is unchecked.

   Do not start the VSA software manually – the script will do that.
2. Click **Calibrate using VSA** button. This will start an instance of the VSA software and set up the parameters to demodulate the signal that has been configured.
3. On the 'VSA measurement running. Please press OK when Equalizer has stabilized', you should first check the Input range in VSA and then observe the Equalizer stabilizing. If it does not converge, you might have to modify the Equalizer parameters.
4. Once the equalizer is stabilized, click **Ok** to continue in the calibration process. The MATLAB script will read the current equalizer frequency response, display it as a MATLAB plot, download the pre-distorted waveform and turn the equalizer in VSA off.
5. Optionally, you can click the **Calibrate** button again (it will now be labeled **Re-Calibrate**) to further improve the EVM performance.
6. If you make changes to your iqmod parameters, clear the **Correction** check box, click **Download** and then **Calibrate (VSA)**.
7. The connection between the MATLAB script and the VSA software remains intact until you either close VSA or exit MATLAB. So, for consecutive calibration runs, the VSA software will not be launched again, but the already running instance will be re-used.

Dual Polarization Digital Modulation

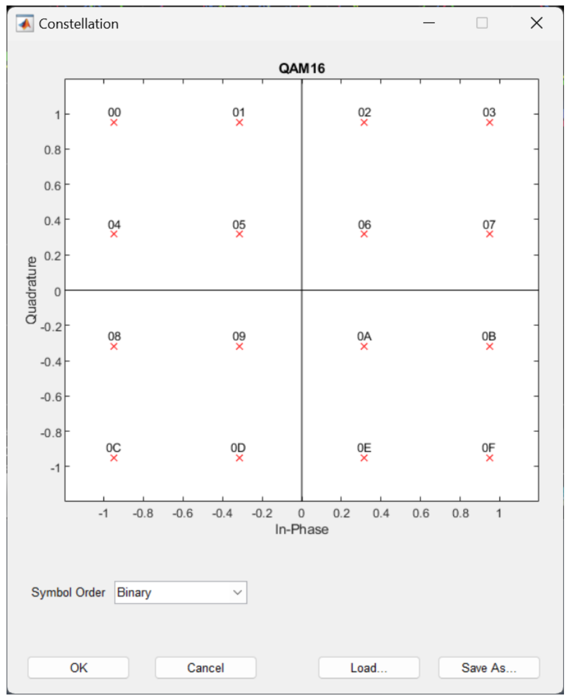
Use this utility to create dual-polarization digital modulation signals. It works very similar to the “Digital Modulation” utility described above but generates two I/Q signals instead of only one. Most of the input fields are identical to the “Digital Modulation” window except:

- **Type of data**: Has a separate popup menu for the X- and Y-polarization. One of the options for the Y-polarization is “Same as X-Pol”.
- **PRBS polynomial/data content/filename**: These fields are duplicated for the Y-polarization. If you choose a different polynomial/data content/filename, note that the same number of symbols will be used for the X- and Y-polarization signal.
- **Shift/Invert**: These fields are also duplicated for the Y-polarization. Note, that the “DC-balanced” check box is NOT duplicated because it would cause a different number of symbols for X and Y, which is currently not supported.
- **# of symbols**: Note that this field is NOT duplicated because the number of symbols for X and Y must (currently) be the same.
- **Quadrature error/IQ Skew/Gain Imbalance**: These fields are duplicated for the Y-polarization.
- **X/Y gain imbalance**: This is a new field that defines the X/Y gain imbalance in dB. Positive numbers amplify X vs. Y. For negative values it is opposite.
- **Download button**: The channel mapping now includes four rows (that is, signals) - IX, IY, QX, and QY.

The “Setup VSA (HW)” button as well as the “VSA → Visualize in VSA” menus behave differently than in the single pol. Digital modulation utility. They set up VSA to capture two signals.

Custom Modulation

Using the “Mapping” button in the “Digital Modulation” or “Dual Polarization Digital Modulation” window, the current modulation type can be visualized and modified.



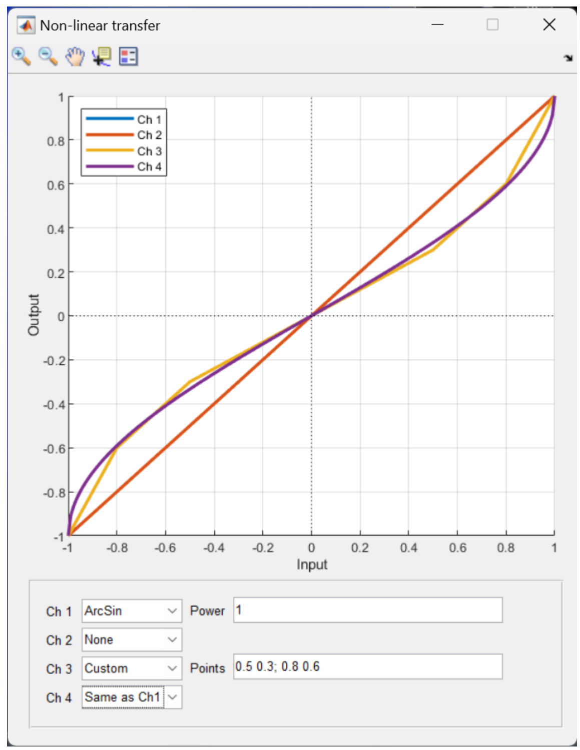
The diagram shows how the symbol values (shown in hexadecimal) are mapped into the complex plane. The following controls are available:

- **Symbol Order**: Defines the sequence in which the symbols are arranged in the complex plane. The symbol order is particularly relevant for square constellations (QAM4, QAM16, QAM64, QAM256, QAM1024). Available options are:
  - **Binary**: Symbols are arranged row by row from left to right.
  - **Grey**: Symbols are arranged, such that neighboring symbols deviate only by a single bit from each other.
  - **Custom**: Allows a user-defined symbol order to be entered in the “Custom Order” field.

- **Custom Order**: Enter a permutation of the numbers from 0 to M-1 (in decimal), where M is the number of constellation points. The sequence of numbers defines the symbol order.
- **Load...**: Loads a (custom) constellation from a file. The file format is ASCII text and is compatible with VSA custom constellation files.
- **Save As...**: Saves the current constellation and symbol order to a file. The file format is ASCII text and is compatible with VSA custom constellation files.

Non-Linear Transfer

When you click the “Show” button next to “Non-linear transfer” in the “Digital Modulation” or “Dual polarization Digital Modulation” window, the non-linear transfer function can be visualized and modified.



The diagram shows the non-linear transfer function for each of the up to four channels. For each channel, the following options are available:

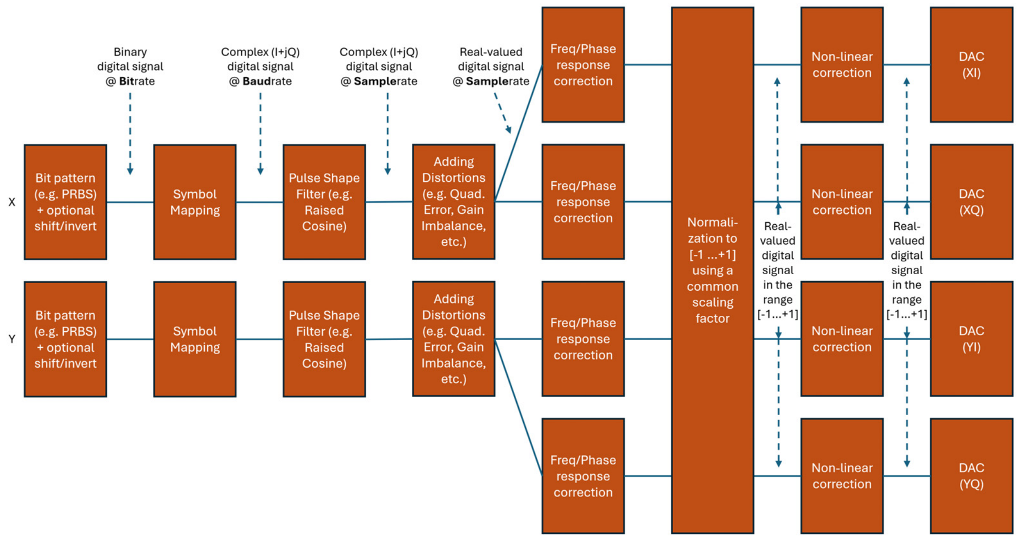
- **None**: Identity transfer function: f(x) = x
- **ArcSin**: f(x) = arcsin(x)/(pi/2)
- **Custom**: Allows a custom transfer function to be defined. Points are entered as x1 y1; x2 y2; x3 y3; ... If the x-coordinates are in the range [0...1], the transfer function is automatically extended into the range [-1...0]. If the “end-points” (that is, x = -1, x = +1) are missing, they are automatically added as (-1,-1) and (+1,+1).

- **File**: Allows a custom transfer function to be retrieved from a file. The file format is ASCII text with two numbers per row representing the X and Y coordinates of the mapping function. The same rules as for "Custom" transfer functions (as described above) applies.
- **Save As...**: Saves the current constellation and symbol order to a file. The file format is ASCII text and is compatible with VSA custom constellation files.

The easiest way to edit a constellation is use "Save As..." of a pre-defined constellation. Edit the file using a standard text editor and the using "Load..." to load the constellation.

Processing Sequence for Digital Modulation

The following diagram shows how a digitally modulated signal is being processed. The algorithm executes the options in the “Digital Modulation” window in a top-to-bottom sequence.

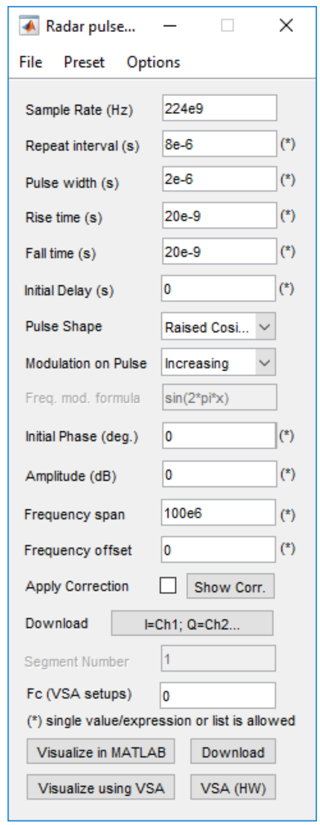


Radar Pulses/Chirps

Use this utility to create a variety of pulsed RF signals. It allows you to generate both RF and Baseband (I/Q) Radar pulses. It directly supports a variety of pulse train arrangements and intra pulse modulation schemes.

- To use the Radar Pulses/Chirps utility, click **Radar Pulses/Chirps** under the **IQ & RF** pane or click "Radar Pulses/Chirps" in the IQ & RF menu of the IQTools main window.

  The following figure shows the **Radar Pulses/Chirps** window:



The following parameters and menu items are available:

- **Repeat interval**: a.k.a. PRI (pulse repeat interval) defines the time from the start of a pulse to the start of the following pulse (see Figure 1, below). If Options > Exact PRI is clear, the specified amount of time will be rounded to a integer number of samples. With Options > Exact PRI selected, the pulse will be repeated as many times as necessary to get an exact PRI timing. If PRI is set to zero, the PRI is set to rise-time + width + fall-time. This is useful, for example, for FMCW signals.

  Multiple values can be specified (separated by space or comma) to generate a sequence of pulses with different PRIs.
- **Pulse width**: Defines the duration of the pulse(s) excluding the rise and fall time (see Figure 1, below). Multiple values can be specified (separated by space or comma) to generate a sequence of pulses with different pulse widths.
- **Rise time**: See Figure 1, below. Multiple values can be specified (separated by space or comma) to generate a sequence of pulses with different rise times.
- **Fall time**: See Figure 1, below. Multiple values can be specified (separated by space or comma) to generate a sequence of pulses with different fall times.
- **Initial Delay**: See Figure 1, below. Multiple values can be specified (separated by space or comma) to generate a sequence of pulses with different initial delay times.

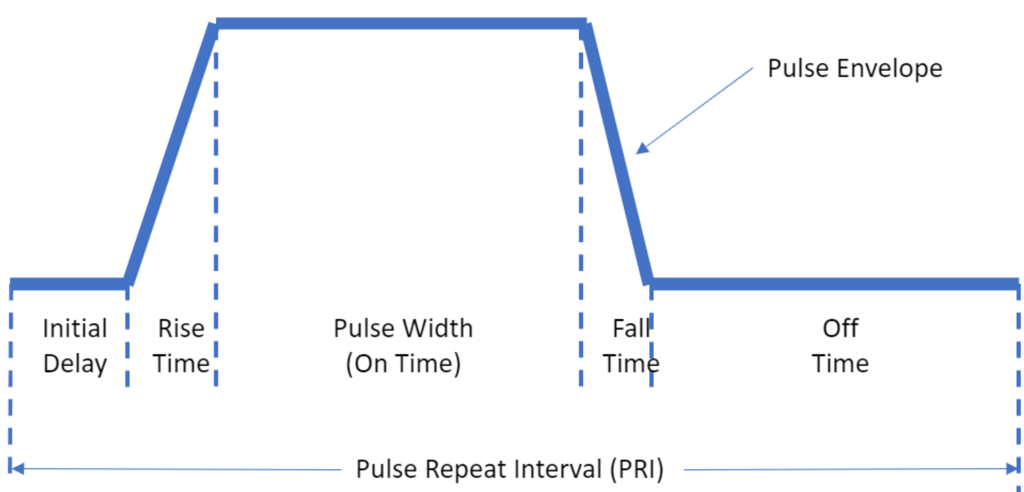


Figure 1 Pulse timing

- **Pulse Shape**: defines the "shape" of the transition times:
  - **Raised Cosine**: the transitions follow the shape of a cosine curve
  - **Trapezoidal**: the transitions are straight lines
  - **Gaussian**: the transitions follow a Gaussian pulse
  - **Zero Signal**: the signal is kept at zero during raise and fall times
- **Modulation on Pulse**: defines the modulation signal during the pulse on time:
  - **None**: no modulation, that is, frequency stays at "Frequency Offset" value
  - **Increasing**: linear frequency ramp from (frequency_offset – ½ frequency span) to (frequency_offset + ½ frequency_span)
  - **Decreasing**: linear frequency ramp from (frequency_offset + ½ frequency span) to (frequency_offset – ½ frequency_span)
  - **V-shape**: Decreasing followed by Increasing frequency ramp
  - **Inverted V**: Increasing followed by decreasing frequency ramp
  - **FMCW**: generates a CW waveform with custom frequency modulation. Since the signal is always "on" in an FMCW signal, the "pulse width", "rise time", "fall time", "initial delay" and "pulse shape" fields have a different meaning: Instead of defining the pulse envelope, they define the how the frequency changes over time:
    - **Initial delay**: specifies the amount of time, the modulation stays as (frequency_offset – ½ frequency_span)
    - **Rise time**: specifies the amount of time, to transition from (frequency_offset – ½ frequency_span) to (frequency_offset + ½ frequency_span)
    - **Pulse Width**: specifies the amount of time, the modulation stays as (frequency_offset + ½ frequency_span)
    - **Fall time**: specifies the amount of time, to transition from (frequency_offset + ½ frequency_span) to (frequency_offset – ½ frequency_span)
    - **Pulse shape**: defines the "shape" of the frequency change (see Pulse Shape above)

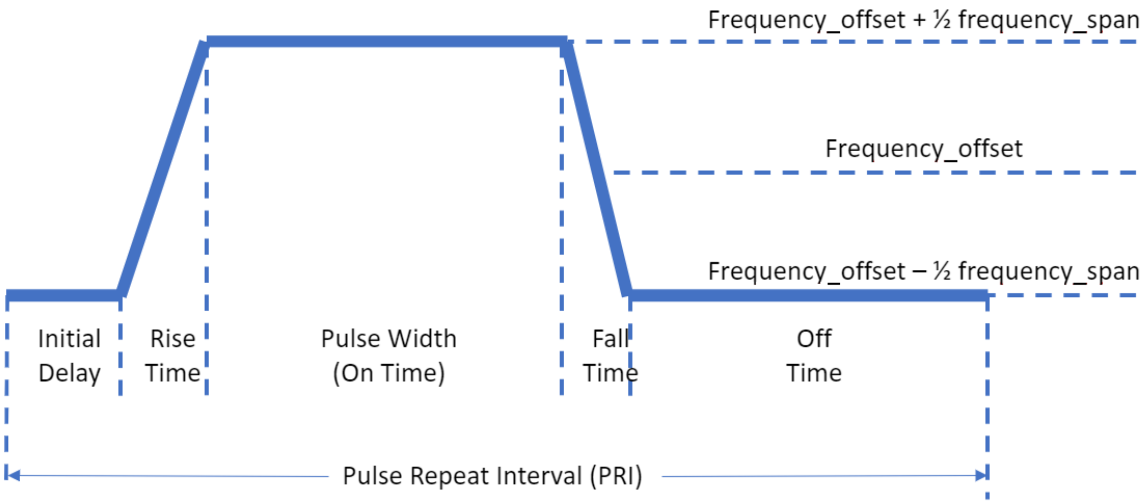


Figure 2 FMCW pulse definition

The best way to understand/view FMCW signals is to look at the spectrogram plot that is generated with "Visualize in MATLAB".

- **Barker-N**: various forms of Barker code (180 degree phase changes at certain points in time).
- **Frank-N**: various forms of Frank code (phase changes of 360/N degrees at certain points in time).
- **User defined**: in this mode, a mathematical formula can be specified that describes the frequency and phase modulation on the pulse. For a description, see the "Freq. Mod. Formula" and "Phase Mod. Formula" fields below. As an example, a non-linear frequency sweep can be implemented using this function.
- **Custom IQ**: in this mode, the user can specify a baseband waveform that is used as a modulation on pulse. For a description on how a custom IQ modulation is specified, see the "Custom IQ waveform" fields below. As an example, a multi-tone modulation on pulse can be generated using this function.
- **Custom Phase Code**: in this mode, the user can specify a sequence of phase values that are equally distributed across the pulse width. See the "Phase Mod Formula" field for a description on how values can be entered. As an example, pulses with arbitrary Barker or polyphase modulations can be generated.

- **Freq. Mod. Formula**: Used when Modulation on Pulse equals “User defined”. Enter a MATLAB expression with "**`x`**" as an independent variable. The expression will be evaluated with **`x`** set to a row-vector with values in the range [0...1). The expression should return a row-vector with the same length as **`x`** and values between -1 and 1 to indicate a frequency deviation from -span/2 to span/2. In case of multiple pulses, the variable "**`i`**" contains the pulse number starting from 1. Example: **`sin(2*pi*x)`** will generate a sine-wave shaped FM pulse with FM deviation equal to frequency_span. If you want no FM modulation at all, put a zero in this field.
- **Phase Mod. Formula**: used when Modulation on Pulse equals “User defined” or “Custom Phase Code”.

  For the **user defined** modulation, enter a MATLAB expression with "**`x`**" as an independent variable. The expression will be evaluated with **`x`** set to a row-vector with values in the range [0...1). The expression should return a row-vector with the same length as **`x`** and values representing the phase in degrees. In case of multiple pulses, the variable "**`i`**" contains the pulse number starting from 1. Example: **`floor(x*4)*45`** generates a pulse with four 45 degree phase steps within the pulse. If you want no PM modulation enter zero in this field.

  For **custom phase code** modulation, the format can either be the same as above or simply a list of phase values in square brackets. Phase values are specified in degrees in this case. Example: **`[0 45 90 135]`** generates the same phase modulation as the one above.
- **Custom IQ Waveform**: Used when Modulation on Pulse equals “Custom IQ”. In this mode a MATLAB expression can be specified that evaluates to a (complex-valued) signal, which is used as a modulation on pulse. The variable “sampleRate” can be used in the expression. If the resulting waveform is shorter than needed, it is replicated; if it is longer than needed, it is truncated.
- Example: to generate a 3-tone modulation on pulse, enter the following MATLAB expression into the Custom IQ Waveform field:
- `iqtone('sampleRate', sampleRate, 'tone', [100e6 200e6 300e6])`

  The IQTools function iqtone is used in this case to generate the multi-tone signal. Any other MATLAB function can be used as well. After applying the custom modulation function, the result is shifted by the value in the “Frequency offset” field.
- **Amplitude**: if a sequence of multiple pulses is generated, this field can be used to specify the relative magnitude of each pulse in dB.

- **Frequency span**: specifies the frequency range(s) of modulation with Increasing/Decreasing/V-shape/Inverted V/User defined modulations on pulse. Multiple values can be specified (separated by space or comma) to generate a sequence of pulses with different frequency spans.
- **Frequency offset**: specifies the center frequency for the modulation on pulse. If frequency offset is zero, an IQ baseband pulse is generated. If frequency span is also zero, only the pulse envelope is generated. Multiple values can be specified (separated by space or comma) to generate a sequence of pulses with different center frequencies.
- **Fc (VSA setup)**: Without external up-conversion, the value should be identical to “Frequency offset”. With external up-conversion, this value should be set to the center frequency that VSA should measure at. Typically, this is the frequency of the LO plus or minus the Frequency Offset.
- **Visualize using VSA**: calculates the pulse waveform and sends it directly to VSA (without going through hardware). It sets up VSA to visualize the pulse(s).
- **VSA (HW)**: calculates the pulse waveform, downloads it into the AWG and sets up VSA to visualize the pulse that is captured from hardware.

Generating a pulse sequence

In addition to repeating the same pulse over and over, the IQTools pulse utility allows you to generate a sequence of pulses with different PRI, pulse width, rise/fall time, center frequency, and so on.

This can be achieved by specifying multiple separated by spaces or comma in one or more of the following fields: PRI, pulse width, rise time fall time, initial delay, amplitude, frequency offset, frequency span.

The number of pulses that will be generated is equal to the maximum number of values in these fields. The values for the other fields will be repeated as many times as needed to achieve the same number of values.

Example: PRI contains three values (8 us, 6 us, 4 us), amplitude contains five values (0, -2, -4, -6, -8), frequency offset contains two values (1G, 2G) and frequency span contains only one value (100M). In this case, five pulses are generated, because five is the maximum of (3, 5, 2, 1). The PRIs for the five pulses will be 8 us, 6, us, 4 us, 8, us, 6 us. The offset frequencies will be 1G, 2G, 1G, 2G, 1G and the frequency span will be 100M for all five pulses.

See also

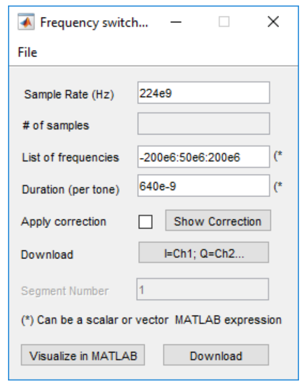
The “Multi-Emitter Pulse/AM/FM” tool can also be used to generated sequences of pulsed signals. The flexibility in terms of parameters is not as large as in the “Radar Pulses/Chirps” tool. However, the Multi-Emitter Pulse/AM/FM tool allows you to generate overlapping pulse sequences and add continuous wave AM and/or FM modulated carriers to the generated signal.

Frequency Switching

This option allows frequency switching between two or more frequencies with adjustable tone duration and phase continuous frequency switching.

- To use the Frequency Switching utility, click **Frequency Switching** in the **IQ & RF** menu of the main window.

  The following figure displays the **Frequency Switching** parameters:



The following parameters are available:

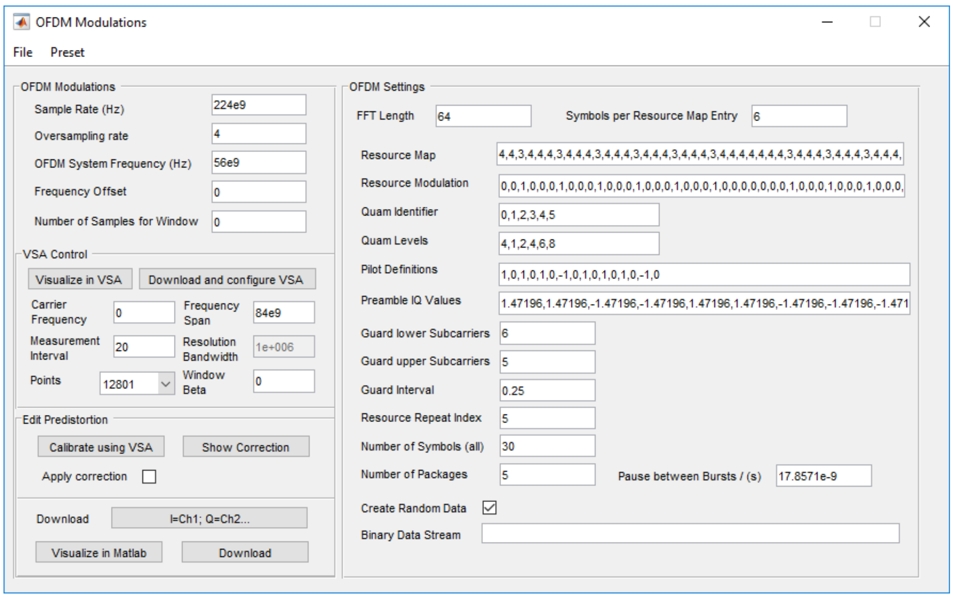
- **List of frequencies**: You can specify the MATLAB expression that evaluates to a vector with frequencies. Example, a list of values separated by commas or space or an expression such as: `start:incr:stop` or `linspace (start,end,numPoints)`.
- **Duration**: specifies the duration for each tone. This can either be a single value or a list of values or a MATLAB expression that evaluates to a vector.

OFDM

This feature enables signal generation with custom modulation schemes.

- To use the OFDM utility, click **OFDM** under the **IQ & RF** pane.

  The following figure shows the **OFDM modulation** window:

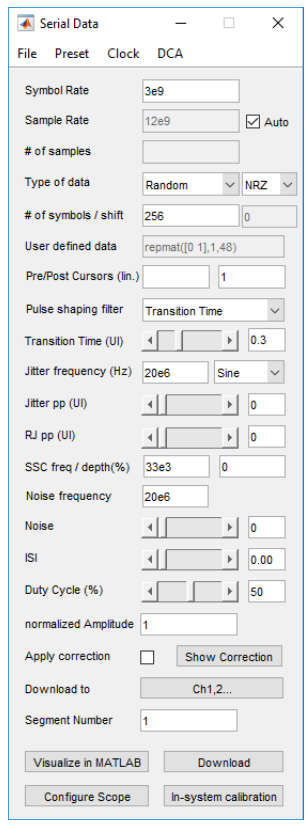


Serial Data Generation

The serial data utility allows you to generate digital (NRZ) or multi-level (PAMx) serial signals with adjustable transition times with or without additional distortions.

- To use the Serial Data utility, click **Serial data generation** under the **Digital** pane.

  The following figure shows the **Serial Data** window:



The following parameters are available in the **Serial Data** utility:

- **Symbol Rate**: Allows you to set the baud rate of the signal. See "Relationship between symbol rate, number of symbols, sample rate and number of samples" on page 108 on how symbol rate, sample rate and number of symbols are related to each other.

- **Sample Rate**: When the “Auto” check box is clear, the sample rate of the AWG can be set. Alternatively, the IQTools algorithm will choose a sample rate that works with the selected symbol rate and number of symbol when the Auto check box is selected. See “Relationship between symbol rate, number of symbols, sample rate and number of samples” on page 108 on how symbol rate, sample rate and number of symbols are related to each other.
- **# of samples**: the number of samples is calculated as follows:

  Number_of_samples = symbol_rate * oversampling_ratio * K

  where, <K> is selected such that the number_of_samples matches the granularity requirements of the AWG. See “Relationship between symbol rate, number of symbols, sample rate and number of samples” on page 108 on how symbol rate, sample rate and number of symbols are related to each other.
- **Type of data**: A pop-up menu to select the type of data content. It works in conjunction with the encoding pop-up menu next to it, which contains a choice of NRZ and PAM4. In the Type of data field, you can choose from the following:
  - **Random**: Uses the MATLAB “rand” function to generate a random set of symbols. The Encoding field determines if NRZ or PAM4 symbols are generated.
  - **Clock**: Uses alternating ‘0’ and ‘1’ as a sequence of symbols.
  - **MLT-3**: Generates an MLT-3 sequence of symbols (see https://en.wikipedia.org/wiki/MLT-3_encoding).
  - **PAM-n**: (n = 3...64) generates a random sequence of PAM-n symbols. The MATLAB “rand” function is used to generate the random sequence of symbols. PAM-4 is an exception: When PAM-4 is selected, the Type of data field switches to PRBS 2^11-1 with the encoding set to PAM-4.
  - **PRBS 2^n-1**: Selects a standard pseudo-random bit sequence as a data source. Depending on selected encoding scheme (NRZ or PAM-4), the PRBS bits are treated as symbols (NRZ) or pairs of PRBS bits are encoded as PAM-4 symbols via the “User-defined levels” field described below. PRBS2^n-1 (BERT) uses the polynomials as defined in M8000 Series User Guide.
  - **Doublet**: Generates random Manchester code symbols.
  - **JP03B**: Generates a stress pattern that consists of 15 times 1-0, followed by 16 times 0-1.
  - **LinearityTestPattern**: Generates a PAM-4 stress test pattern: 0 1 2 3 0 3 0 3 2 1.

  - **SSPRQ**: Generates a PAM-4 stress pattern, see https://www.ieee802.org/3/bs/public/adhoc/smf/16_04_19/anslow_01_0416_smf.pdf.
  - **PRBS13Q gray coded**: Generates the PRBS13Q stress pattern.
  - **QPRBS13**: Generates the QPRBS13 stress test pattern.
  - **QPRBS13 RZ**: Generates the QPRBS13 stress test pattern with every other bit set to zero (that is, lowest PAM-4 level).
  - **QPRBS13 R1/2**: Generates the QPRBS13 stress test pattern with every other bit set to the (high level + low level)/2.
  - **Dual PAM4**: Generates the sum of two random PAM4 patterns, effectively generating a PAM-7 signal, but with non-equal probability of each level.
  - **Custom PAM4 LFSR**: Generates a custom pattern. The pattern itself can be modified in the code (see iqmod.m, line 918).
  - **SSPR (OIF-CEI-4.0)**: Generates the stress test pattern defined in OIF-CEI-4.0.
  - **Pattern from file**: Allows you to enter the name of a file with symbol values in it. The values in the file are expected to be numbers from 0 to M-1 (where M is the number of symbols in the selected modulation format) or 0, 1/(M-1), 2/(M-1), ..., 1. As described in “User defined data” further in this section, intermediate levels are also possible.
  - **User defined**: Allows you to enter a sequence of symbol values. The bits are converted to symbols. For details on the syntax, see “User Defined data” further in this section.
- **# of symbols**: specifies the number of symbols to be generated. When selecting a PRBS as type of data, this field will be pre-populated with the “native” length of the PRBS (or an integer multiple thereof). You can overwrite the number of symbols, for example, if you want a PRBS 2^N instead of 2^N-1.

  In case of “Pattern from file” as data type, the number of symbols will be populated by the number of symbols read from the file. See “Relationship between symbol rate, number of symbols, sample rate and number of samples” on page 108 on how symbol rate, sample rate and number of symbols are related to each other.
- **Shift**: when type of data is set to PRBS, this value will “shift” the PRBS pattern by the given number of symbols. This can be useful to generate, for example, two PRBS patterns with the same length on two AWG channels, but have them “misaligned” by a certain number of symbols.

- **User defined data**: The User defined data field is only visible if Type of Data is set to “User defined”. It contains a MATLAB expression that evaluates to a vector of “symbol” values. The values must be in the range 0....1, with 0 representing the lowest voltage level and 1 the highest voltage level. For NRZ signals, the field typically contains a vector of 0’s and 1’s. For PAM4, the nominal values are 0, 1/3, 2/3 and 1.

You can either fill in a list of values separated by spaces or comma (such as, 1 0 0 1 1 1 0 1 0 0 1) or a MATLAB expression that evaluates to a vector of values (such as, to generate a vector of 256 random 0’s and 1’s, you can put in the expression: randi([0 1], 256, 1)). To make PAM4 signals easier to read, it can be written as 1/3*[0 1 2 3 ...], instead of 0, 1/3, 2/3, 1 ...

In addition to the “nominal” values (0 & 1 for NRZ; 0, 1/3, 2/3, 1 for PAM4), you can also use any fractional value between 0 and 1 to represent intermediate voltage levels. For example, 0, 0, 0, 0.8, 0, 0, 0 can be used to generate an isolated “1” that does not quite reach the correct voltage level.

To simplify generating PRBS sequences with errors in them, the **User defined data** field behaves as follows:

  - Whenever you change the selection in the **Type of Data** drop-down menu, the corresponding data pattern is copied to the **User defined data** field. (For example, when you select PRBS2^7-1, PAM4 as the **Type of data**, the **User defined data** field changes to “1/3 * [1 2 2 3 2 2 3 1 2 3 3 2 3 0 1 3 2 3 3 1 3 0 3 3 2 0 0 1 1 2......]”).
  - If you change the selection to **User defined data** afterwards, that same data pattern will be generated. Now you have the possibility to change individual symbols by changing one of the numbers in the square brackets to another number in the range 0...3 or even force the signal to go through the middle of an eye by changing one of the numbers to a fractional value between 0 and 3.

- **User pattern file**: This field is only visible when type of data is set to “Pattern from file”. Supported file formats are - *.TXT*, *.CSV,* and *.PTRN*. TXT and CSV format are expected to contain a sequence of symbol values separated by spaces or newlines. The same values as described in “User defined data” described above are supported. The *.PTRN* format is a legacy file format from BERT pattern generators.

- **User defined levels**: In this field, the mapping of symbol values to output voltage levels can be defined, such as, gray coding or linear coding. The number of values must match the number of symbol values. The values must be in the range [0...1]. For PAM4, the default setting for this field is "0 1/3 1 2/3" (gray coding). This means that symbol value 0 is mapped to 0, symbol value 1 is mapped to 1/3, symbol value 2 is mapped to 1, symbol value 3 is mapped to 2/3. By changing the order of the values, the coding can be changed (0 1/3 2/3 1 is linear mapping). This field can also be used to generate non-equidistant levels (for example, 0 0.2 0.5 1).
- **Pre-/Post cursors**: These two fields can be used to define the linear coefficients for pre-/de-emphasis of the signal. An arbitrary number of pre- and post-cursors can be specified. The first cursor in the "Post cursor" field is treated as the "main" cursor. Applying the cursors can be viewed as a convolution of the serial data signal (symbol) with the coefficients (c).

  Example: with one pre-cursor (c[-1]), the main cursor (c[0]) and two post-cursors (c[1] and c[2]), the following formula is used to calculate each new symbol values:

  new_symbol[n] = symbol[n-1]*c[-1] +
  symbol[n]*c[0] +
  symbol[n+1]*c[1] +
  symbol[n+2]*c[2]

  As an example, cursor values -0.2, 1.3, -0.1 cause the following pre-distortion of the signal:

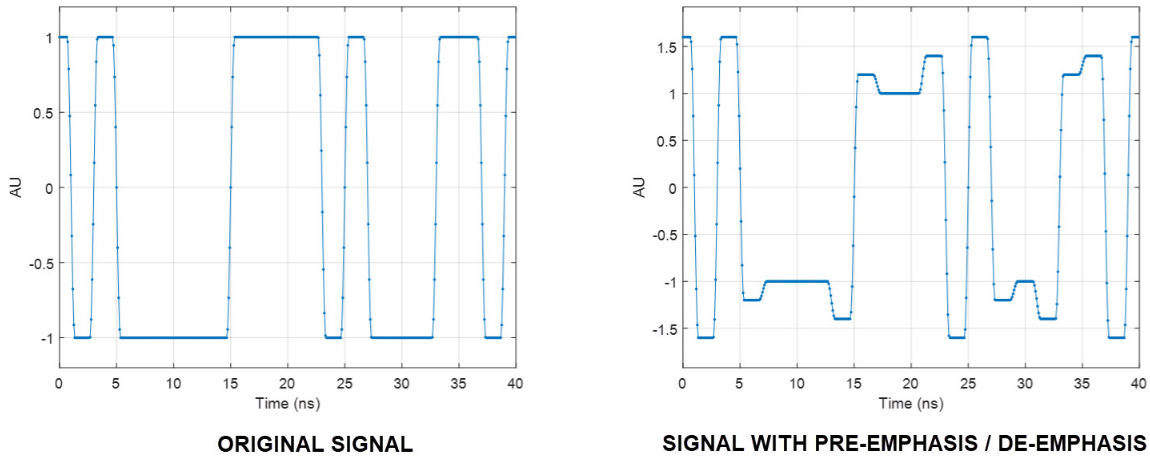


Figure 3 Change in signal form after applying cursor values

If the result of the convolution exceeds the DAC range, the signal is automatically scaled back to the full DAC range in order to avoid clipping.

- **Pulse shaping filter**: The selection in this field determines which algorithm is used to translate the sequence of digital symbols into a waveform that can be played back by the AWG. Unlike a pattern generator, where one symbol per clock cycle is generated, an AWG typically uses more than one sample per symbol to generate a digital signal. The oversampling ratio (sample_rate/symbol_rate) does not need to be an integer value.

Figure 4 Effects of applying pulse shaping filter

The following options are available as a pulse shaping filter:

- **Transition time**: With this selection, the waveform is mathematically constructed from cosine-shaped transition segments and straight-line segments. The duration of the transition segments is controlled by the transition time parameter. The samples are placed on this “ideal” waveform at the distance of a sample interval (= 1/sample_rate):

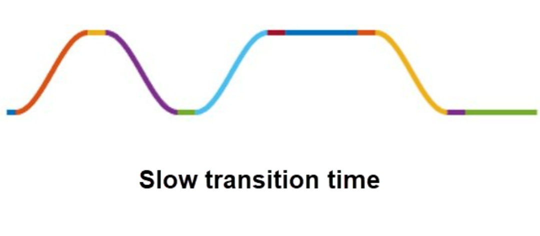


Figure 5 Effects of applying transition time

Choosing “Transition time” works best with large oversampling values (that is, low baud rate relative to sample rate), and the transition time parameter set to value large enough so that the algorithm can place at least 3 or 4 sample points on the transition.

- **Raised cosine/Root-raised cosine (RC/RRC)**: With this selection, the waveform is constructed by adding up the impulse response of a RC resp. RRC filter which is shifted to the symbol position and scaled with the symbol value.

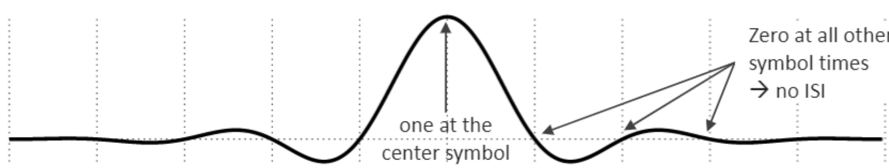


Figure 6 Impulse response of an RC/0.3 filter

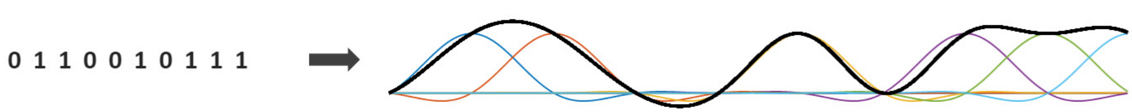


Figure 7 Adding up shifted copies of the impulse response yields the final signal

The benefit of using RC/RRC pulse shape is that the bandwidth of the generated signal can be minimized (see “Alpha” below). This method is best suited for high baud rate signals with only a few samples per symbol. However, with less than approximately 1.5 samples per symbol, this method also reaches its limit.

NOTE Both algorithms (transition time and RC/RRC) work with NRZ signals as well as multi-level (for example, PAM-4) signals.

- **Transition Time**: this field is only visible when Pulse Shape Filter is set to Transition Time. As described above, this field defines the duration of a transition segment as relative to the symbol duration or unit interval (UI).

  Note, that this parameter describes the “0 to 100%” transition time. The 20/80 resp. 10/90 transition times will be shorter.

  The background of this field turns yellow if fewer than four samples can be placed on the transition. In that case, you can still get a valid signal, but it may be distorted.

  It is possible to specify rise and fall time to be different values by putting two values into the “Transition Time” field, separated by a space or comma. Example: 0.3 UI rise time, 1 UI fall time:

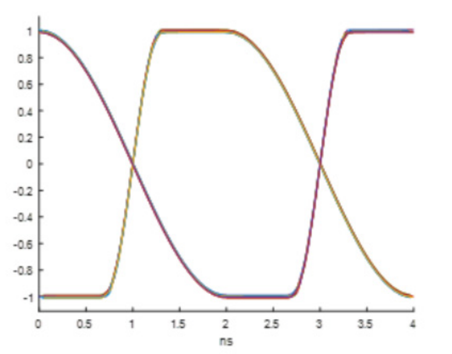


Figure 8 Effect of transition time on rise and fall time

- **NSym/Alpha**: These two fields are only visible when Pulse Shape Filter is set to Raised Cosine or Root-Raised Cosine. The **NSym** field defines the length of the RC/RRC filter in number of symbols. For smaller values of Alpha, a larger NSym should be chosen because the “wiggling tails” of the filter reach out further. IQTools populates this field with a reasonable default value depending on the Alpha value. However, this value can be overwritten. Note, that NSym must be less than (number_of_symbols – 5).

  In the **Alpha** field, the roll-off factor of the RC/RRC filter can be defined. The alpha value must be greater than zero and less or equal to 1. The value of Alpha determines the bandwidth of the generated signal. The nominal bandwidth can be calculated as: symbol_rate * (1 + alpha)/2.
- **Jitter frequency and shape**: These parameters define the frequency of added periodic jitter and the shape of the jitter function. Note, that the jitter frequency may be rounded such that an integer number of jitter periods fits into the generated signal. Example: if the signal is 1 us long and a jitter frequency of 20.1 MHz is selected, it will be rounded to 20 MHz, since **`round(1 us * 20.1 MHz)/1 us = 20 MHz`**.
- **Jitter pp**: Peak-peak deviation of the periodic jitter. The value is given as a fraction of the unit interval (UI). A value of 1 (or larger) will cause the eye diagram to be completely closed. If this field is zero, no periodic jitter will be added.
- **RJ pp**: Peak-peak deviation of pseudo-random jitter. Just like periodic jitter, this pseudo-random jitter is calculated into the signal. Since the signal is played back repetitively, so is the jitter. So, it is in fact not truly “random”. The jitter function is similar to a gaussian deviation, but with a peak value that is limited at 6 times the RMS value.
- **SSC frequency and depth**: these parameters define the frequency and depth of spread-spectrum clocking function. Note that the SSC frequency may be rounded such that an integer number of periods fits into the generated signal. Example: if the signal is 100 us long and the SSC frequency is set to 33 kHz, the frequency is rounded to 30 kHz, since **`round(100 us * 33 kHz)/100 us = 30 kHz`**. The **SSC depth** is given in percent of a unit interval (UI). Example: If the SSC depth is set to 2%, the period-to-period deviation is 2% of a UI when the SSC function is at its maximum value. Note, that the absolute timing deviation is usually much larger than the SSC depth because the absolute deviation is the cumulative sum of the period-to-period deviations.

The SSC shape is currently fixed at “Triangle”, but this can be modified in the code, if desired (iserial.m, line 524).

- **Noise frequency**: defines the frequency of level noise. Note that the level noise frequency may be rounded such that an integer number of periods fits into the generated signal. Example: if the signal is 1 us long and the noise frequency is set to 20.1 MHz, the frequency is rounded to 20 kHz, since `round(1 us * 20.1 MHz)/1 us = 20 MHz`.
- **Noise**: defines the amplitude of level noise. It is given as a fraction of the signal amplitude, that is, if the noise amplitude is set to 1 (or larger), the eye will be completely closed. If the noise amplitude is set to zero, no level noise will be added.
- **ISI**: This is a “quick-but-unpreferred” method of adding some amount of inter-symbol interference (ISI). It is implemented by running the signal through the following IIR filter:

  new_symbol[n] = (1 – isi) * old_symbol[n] + isi * new_symbol[n-1]

  The ISI value must be greater or equal to 0 and less than 1. To achieve some effect, the value must be close to, but not equal to 1. For example, 0.95. The amount of ISI that will be generated is difficult to quantify. Therefore it is not recommended to use this function.

  A more accurate method to add ISI is to specify an S-parameter file in the correction management window.
- **Duty cycle**: for NRZ signals, defines the duration of high level in a “1 0” pattern. Nominally, the duty cycle is 50%, that is, the signal is 50% at high level and 50% at low level in a clock pattern. With a duty cycle less than 50%, the duration of high level is reduced and the crossing point in the eye diagram moves towards low level.

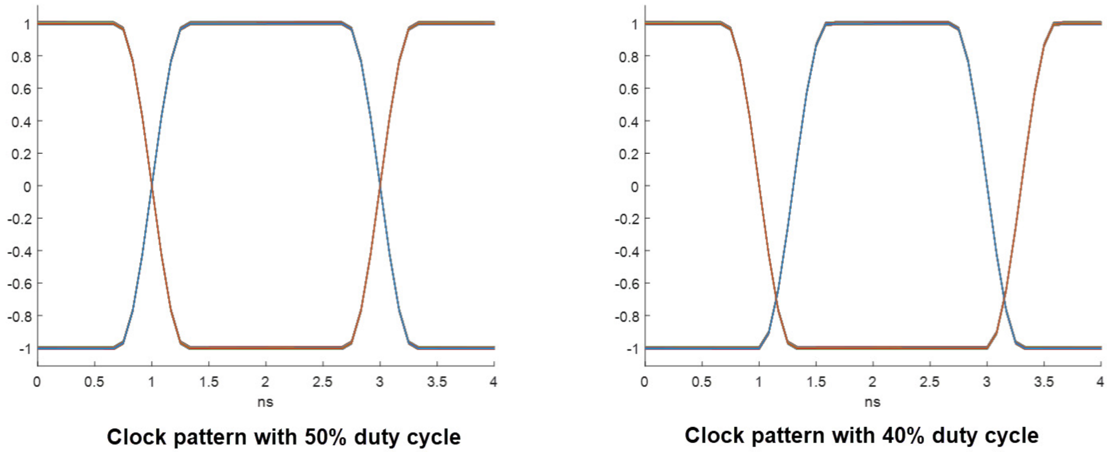


Figure 9 Effect of duty cycle on clock pattern

- **Normalized amplitude**: This parameter defines which fraction of the full-scale DAC amplitude will be used by the signal. Normally, this field should be set to 1, meaning that the full DAC swing is being used. Under special circumstances, a value less than 1 might make sense. For example, if you want to define multiple waveform segments with different amplitudes which are played back as part of a (hardware) sequence. In all other cases, it is better to adjust the amplitude of the signal using the amplitude setting of the AWG (such as, through the "amplitude" parameter in the configuration window)
- **File > Save Data Pattern**: In addition to the "Save Waveform..." option, the serial data utility also provides an option to save the data pattern (that is, the sequence of symbols). This can be useful to analyze/modify the data pattern externally and then load it using file type "Pattern from file".

Relationship between symbol rate, number of symbols, sample rate and number of samples

In most cases, the IQTools algorithm takes care of calculating the desired waveform given the parameters in the serial data window. Occasionally, that is not possible and an error message will pop up. In this case, it is helpful to understand what the limitations are, so that parameters can be modified accordingly. The parameters symbol rate, number of symbols, sample rate and number of samples are related to each other through the following formula:

$$\frac{Number\ of\ samples}{Sample\ Rate} = \frac{Number\ of\ symbols}{Symbol\ Rate}$$

Both the left and right-hand side of this equation represent the overall signal duration. On top of that, most AWGs impose a certain granularity requirement on the number of samples in a waveform (for example, the number of samples must be a multiple of 512 for the M8195A). If you choose fixed values for sample rate, symbol rate and number of symbols, it is very likely that the resulting number of samples does not meet the granularity requirement. Normally, the algorithm takes care of that be replicating the waveform a sufficient number of times to meet the granularity requirement. But sometimes, this may not be possible because it exceeds the available amount of memory. In this case, either the sample rate or the number of symbols must be modified.

If the sample rate is fixed (that is, the “Auto” check box next to sample rate is not selected), IQTools will modify the number of symbols, otherwise, IQTools will modifty the sample rate in order to meet the granularity requirements.

Example: Using the M8195A, we want to generate a PRBS 2^15-1 at 25 GBaud. Initially, using a sample rate of 64 GSa/s in 4-channel mode with only 256 kSa/channels available.

The number of symbols is 2^15-1 = 32767. Using the formula above, the number of samples is calculated as 32767 * 64G/25G = 83883.52 samples. This is obviously not possible because an AWG cannot generate a fractional number of samples. Repeating this waveform 25 times makes it an integer number: 25 * 83,883.52 = 2,097,088. But this number is still not divisible by 512. To meet this requirement, the signal must be repeated a total of 200 times, which makes it 200 * 83883.52 = 16,776,704 samples. Now this number by far exceeds the available memory size of 256 kSa.

There are several possibilities to work around this situation:

- i Selecting the “Auto” check box next to Sample Rate allows the IQTools algorithm to choose a “convenient” sample rate. With that, IQTools rounds the “exact” number of samples (83,883.52) to the nearest multiple of the granularity (83,968 samples) and adapts the sample rate to 64.064455 GSa/s, such that the desired symbol rate is generated. The 83,968 samples easily fit into the 256kSa memory.

ii Depending on the intended measurement, it may not be mandatory to generate exactly 32767 (= 2^15-1) symbols. For example, in an eye diagram measurement, a larger number of symbols does not hurt. In this case, the “Auto” check box can remain clear and IQTools will automatically increase the number of symbols in this particular case to 32800. It also shows a warning dialog informing you that it has modified the number of symbols. In this case, the number of samples turns out to be 83,968 as well, which fits into the 256 kSa memory.

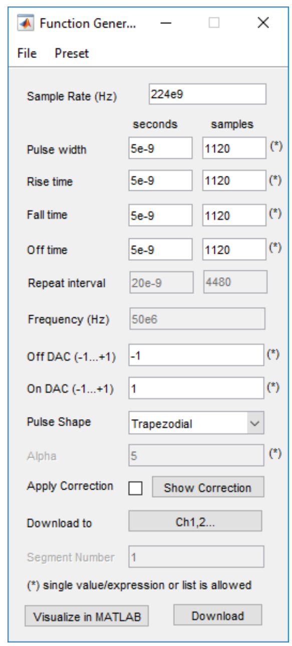
iii If both the exact number of symbols AND the sample rate need of 64 GSa/s have to be maintained, the only other option is to switch to the large memory mode, set the number of symbols to 200*32767 (see formula above) and download the 16,776,704 samples.

Pulse/Function Generator

This utility allows you to generate various types of pulses or a sequence of pulses by defining the timing and level parameters. Unlike the "Radar Pulses and Chirps" utility, where you define RF pulses, this utility only generates the envelope or time-domain pulse.

- To use the Pulse/Function Generator utility, click **Pulse/Function Generator** under the **Digital** pane.

  The following figure shows the **Pulse/Function Generator** window:



The following parameters are available:

- **Pulse Width, Rise time, Fall time, Off time**: These parameters define the timing of the pulse as shown in the following figure.

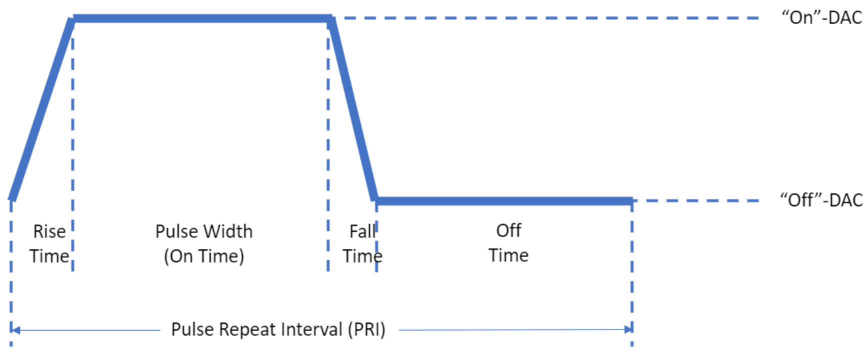


Figure 10 Effect of Pulse Width on the timing of the pulse

- **Repeat Interval**: Is calculated based on the pulse width, rise time, fall time and off-time parameters.
- **Frequency**: Is calculated as 1/repeat_interval.
- **Off DAC, On DAC**: Defines the (normalized) DAC levels for the On-time and Off-time of the pulse (refer to the figure above). A value of 1 refers to the maximum DAC value, -1 refers to the minimum DAC value.
- **Pulse shape**: Selects the "shape" of the transition. The following figure shows a comparison of the available transition shapes:

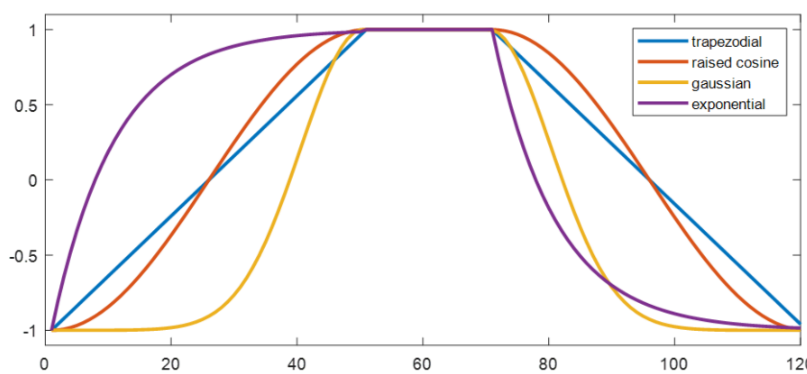


Figure 11 Effect of Pulse Shape on the transition

The "exponential" and "gaussian" shapes can be further influenced using the "Alpha" parameter.

Generating a pulse sequence

In addition to repeating the same pulse over and over, the IQTools pulse utility allows you to generate a sequence of pulses with different pulse width, rise/fall time, off-time, DAC On/Off values.

This can be achieved by specifying multiple values separated by spaces or comma in one or more of the following fields: pulse width, rise time fall time, off-time, On-DAC, Off-DAC. Through the combination of these parameters, it is possible to define quite complex pulse shapes.

The number of pulses that will be generated is equal to the maximum number of values in these fields. The values for the other fields will be repeated as many times as needed to achieve the same number of values.

Example: Pulse width contains 3 values (8 us, 4 us, 1 us), rise/fall time contains a single value (1 us), On-DAC contain 3 values (1, 0, -0.5), Off-DAC contains a two values (-1, 0.5). In this case, 3 pulses are generated, because 3 is the maximum number of values. The resulting pulse looks like this:

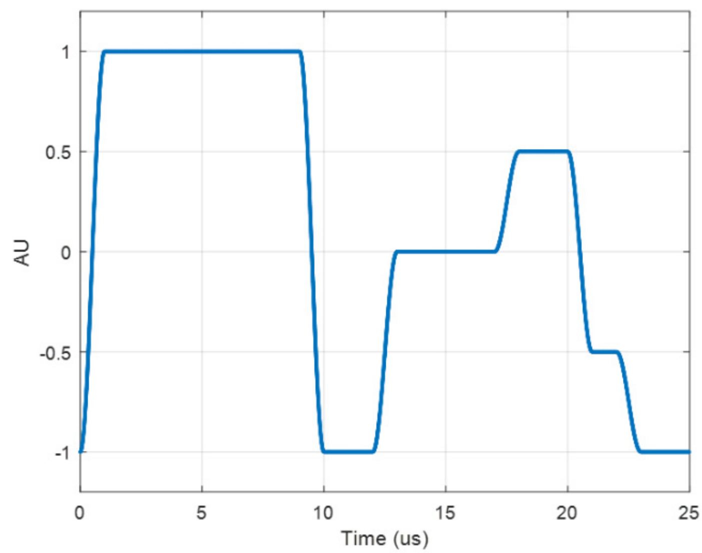


Figure 12 Example of generating a pulse sequence

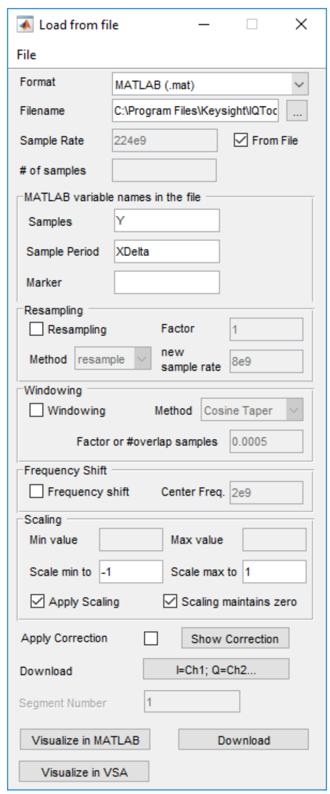
Load Waveform from File

This utility allows you to feed external data to IQ Tools by loading a waveform file. You can select the type to input file to download. For 'CSV', the file must contain one or two columns of data (separated by comma) that are loaded into channel 1 resp. channel 2 of the AWG.

For file type 'MAT', the MATLAB file must contain at least one vector that contains the data. A real data vector will be loaded in channel 1, a complex vector will be loaded in both channels (real to channel 1, imaginary to channel 2). Optionally, the MATLAB file can contain another scaler variable that holds the sampling period. The names of these variable must be specified in the fields available.

- To use the Load Waveform from File utility, click **Load Waveform from File** under the **File** pane.

  The following figure shows the **Load from File** window:



The following parameters are available:

- **Format**: Select amongst the following file formats:
  - **ASCII/CSV**: One or two columns with sample values plus up to four (optional) marker columns. Sample values can either be floating point values in the range [-1 ... +1] or DAC values, for example, -128 to +127. Marker values must be 0 or 1.

    Optionally, the CSV file can contain the sample rate in the first line in the format “Samplerate = 64e9”.
  - **MATLAB**: A “.mat” file that contains at least one variable representing the vector of samples, optionally a second variable representing the sample rate or sample period and optionally a third variable with marker values. The variable names that contain the sample values and the sample rate can be specified in the following panel (Samples/Sample Period/Marker).
  - **16-bit I/Q pairs**: Binary file format with alternating I and Q values as signed 16-bit integers. LSB or MSB is selectable.
  - **16-bit binary**: Binary file format with signed 16-bit integers. LSB or MSB is selectable.
  - **12-bit packed binary**: Binary file format with signed 12-bit sample values in a tightly “packed” structure. In this format, 3 bytes (= 24 bits) are used to represent two 12-bit sample values. The bits are arranged as follows:

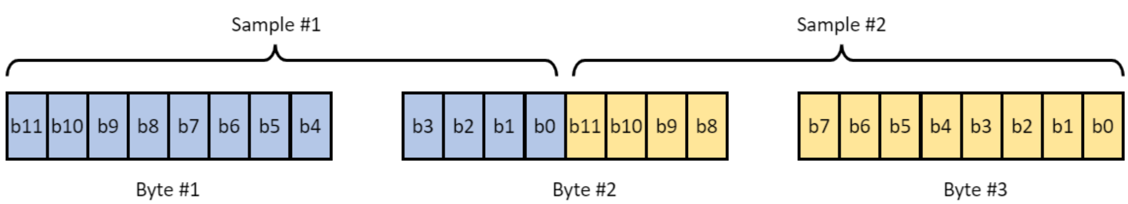


Figure 13 Structure of 12-bit packed binary format

  - **Oscilloscope format**: The file format that is used by Keysight oscilloscopes. Supported formats are *.bin*, *.csv*, *.txt* or *.osc.*
  - **ELT format**: Binary file format with packed 12-bit I/Q values in a tightly packed structure. In this format, 6 bytes (= 48 bits) are used to represent two 12-bit I/Q pairs (= 2 * 2 * 12 bits). The bits are arranged as follows:

I1 b11	I1 b10	I1 b9	I1 b8	I1 b7	I1 b6	I1 b5	I1 b4	I1 b3	I1 b2	I1 b1	I1 b0	Q1 b11	Q1 b10	Q1 b9	Q1 b8

Word #1

Q1 b7	Q1 b6	Q1 b5	Q1 b4	Q1 b3	Q1 b2	Q1 b1	Q1 b0	I2 b11	I2 b10	I2 b9	I2 b8	I2 b7	I2 b6	I2 b5	I2 b4

Word #2

I2 b3	I2 b2	I2 b1	I2 b0	Q2 b11	Q2 b10	Q2 b9	Q2 b8	Q2 b7	Q2 b6	Q2 b5	Q2 b4	Q2 b3	Q2 b2	Q2 b1	Q2 b0

Word #3

Figure 14 Structure of ELT format

- **Sample Rate**: If the “From File” check box is selected, the AWG sample rate is read from the file (CSV with header or MATLAB format). If the “From File” check box is clear, the sample rate must be specified in this field.
- **# of samples** is filled in with the actual number of samples after the “Visualize in MATLAB” or “Download” button is clicked.

Resampling panel

- **Resampling** check box: Enables/disables software re-sampling. Using this functionality, a waveform file that was sampled at a low sample rate can be brought into the sample rate range of the AWG.
- **Resampling Method**: The following methods are available:
  - **resample**: uses the “resample” function that is built into MATLAB. Note, that this function does not protect the beginning and end of the waveform from resampling artefacts (unless windowing is also enabled).
  - **interpolate**: resample the data at a higher rate through lowpass interpolation using the MATLAB function “interp”. Note that this method can only be used with an integer resampling ratio.
  - **FFT**: It works by first calculating the FFT of the waveform, then extending the frequency domain signal with zeros followed by an inverse FFT. This method works well if the original waveform is periodic. The periodicity is preserved. This method uses the MATLAB function “interpft” to perform resampling. This method works well for periodic waveforms and retains the periodicity. This method might be slow for large waveforms or large resampling factors (>> 10 MSamples).

- **Linear**: Performs linear interpolation between the sample points in the waveform.
- **Arbitrary**: Uses a Keysight-proprietary arbitrary resampling function. It preserves the periodicity of a signal. This algorithm works well, even if the resampling factor is decimal number that cannot be expressed as a fraction with small numerator and denominator.

- **Resampling Factor**: Allows you to specify the factor by which the input data will be oversampled. Both the length of the input waveform and the sample rate will be multiplied by resampling factor to get the final waveform length resp. the resulting AWG sample rate.
- **New Sample Rate**: Shows the AWG sample rate (= input sample rate times resampling factor).

Windowing panel

- **Windowing** check box: turns windowing functionality on or off. Windowing can be used to avoid a spectral glitch when there is a phase discontinuity or a sudden transition between the end of the waveform file and the beginning.
- **Windowing method**: The **Cosine Taper** method smoothly fades out both the beginning and end of the waveform, such that the transition will always be at zero, while the **Crossfade** method creates an overlap between the end and beginning of the waveform, similar to a dissolve operation.

  The following three graphs in Figure 15 show the transition from the end of a waveform (blue) to the beginning of the waveform (red) under different windowing methods. In this example, there is both a level as well as a phase discontinuity in the original waveform.

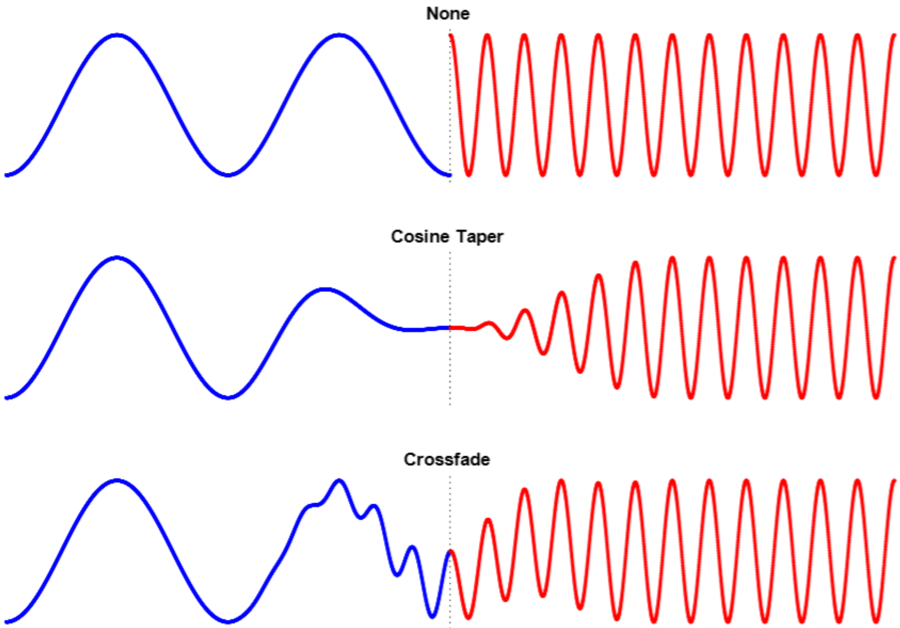


Figure 15 Effect of various windowing methods

- **Factor or #overlap samples**: Defines the length of the overlap resp. fading range of the waveform. It can either be given as a fraction of the waveform length (for example, “0.01” means 1 percent of the waveform length) or in number of samples (for example, a value of 1000 means an overlap/fade area of 1000 samples).

Frequency shift panel

- **Frequency shift** check box enables/disables frequency shifting. In mathematical terms, frequency shifting means multiplying the input waveform with a sine wave of the desired frequency. This operation is performed with complex-valued numbers, such that an I/Q waveform remains intact.
- **Center frequency** is the frequency by which the input signal is shifted. The center frequency can also be negative, which causes a shift towards lower (or even negative) frequencies.

  Note that the center frequency is rounded such that an integer number of periods of the shift frequency is generated.

Scaling panel

- **Apply Scaling** check box enables/disables scaling of the input waveform. If it is turned off, waveforms that are smaller than the [-1 ... +1] interval will not use the full DAC range, which causes a degradation of the signal-to-noise ratio. Except for special purposes, this check box should always be selected.
- **Scaling maintains zero**: When enabled, it causes the scaling to be symmetrical around zero, that is, the sample values are simply multiplied by a certain value. If this check box is cleared, the waveform is scaled such that its maximum equals the “Scale max” value and its minimum equals the “Scale min” value. A waveform value of zero may or may not end up at zero after this operation.
- **Scale max/Scale min** defines the maximum/minimum of the waveform values after scaling.

Sequencer Setup

For some of the AWG models (currently M8198A, M8195A and M8190A), IQTools also supports the sequencer that is built into these AWGs.

To set up a sequence, you must first define and download individual waveform segments, which will later be combined to form the complete sequence. Once the segments are prepared, you can define the sequence itself.

The sequence is displayed as a table that specifies:

- which waveform segment to use,
- how many times it should be looped, and
- the condition under which the next segment will follow.

To define the individual waveform segments, use the **Segment Number** field that is available in each of the waveform tools (Multi-Tone, Digital Modulation, Radar Pulse, and so on.)

The default segment number is 1, which means the waveform will be downloaded to segment 1. (Waveform segments are numbered sequentially from 1 up to the model-specific limit of the AWG.)

To create multiple waveform segments:

1. Open one of the waveform creation tools such as Multi-Tone, Digital Modulation, Radar Chirps, and so on.
2. Define the desired waveform parameters.
3. Set the **Segment Number** to **1** and click **Download**.
4. Modify the parameters (or select a different waveform tool), set the **Segment Number** to **2**, and click **Download** again.
5. Repeat this process until you have created all the waveform segments needed for your sequence.

If you want to generate the same type of waveform with different parameters, you can open multiple instances of the same tool by clicking its button again in the IQTools main window. The current parameters are automatically copied into the new instance for easy modification.

NOTE

The order in which you download waveform segments determines the order in which they are written to the waveform memory in the hardware. Depending on the AWG model, the sequencer has certain limitations concerning the order in which waveform segments have to be located in memory.

For detailed information about these model-specific restrictions, refer to the User Guide of the respective AWG.

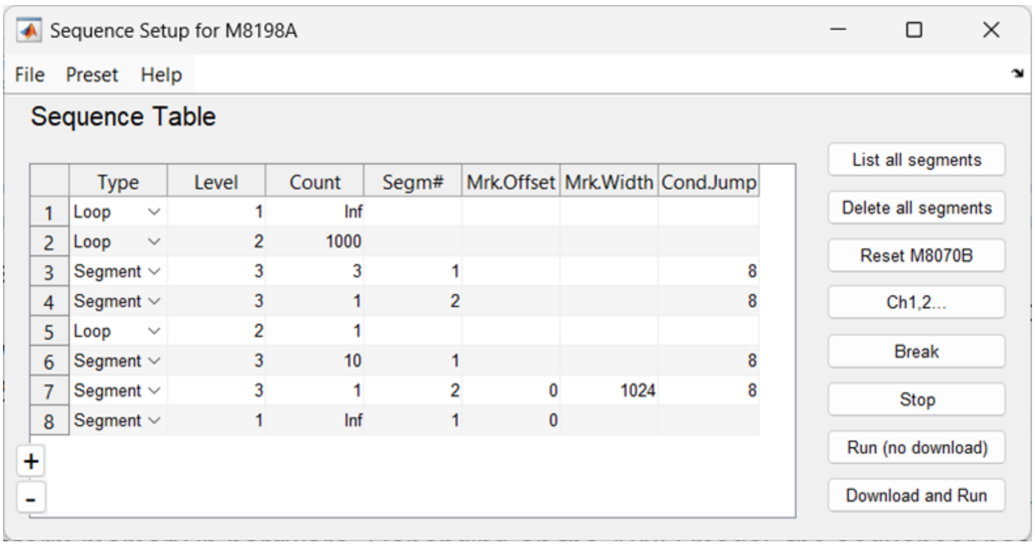
Once all the waveform segments have been defined and downloaded, open the **Sequencer Setup** tool from the **Sequencer** panel of the IQTools main window. The sequencer capabilities vary depending on the AWG model, and therefore, the **Sequencer Setup** window may look and behave differently based on the connected instrument.

NOTE

Note that the **Sequencer Setup** window in IQTools might not offer the full set of capabilities of the sequencer.

Sequencer Setup for M8198A

The sequencer in the M8198A supports nested loops, independent sequences per channel, marker control with variable offset and width as well as conditional jumps.



You can insert and delete rows from the table with the "+" and "-" buttons on the bottom left of the **Sequence Table**. In each row of the **Sequence Table**, you can specify the following:

- **Type** – can be "Loop" or "Segment". As the name suggests, "Loop" defines a loop that consists of two or more segments or inner loops on a higher nesting level. A "Segment" is simply a looped segment with no further nesting levels underneath
- **Level** – defines the nesting level of a loop or a segment. (In programming jargon, think of "Level" as the amount of indentation). A Loop on level N spans all the following lines with Level > N. The next Loop or Segment on Level <= N follows sequentially. In the example above, the loop on line 1 (Level 1) spans from line 2 to 7, since they are all on levels > 1. The loop on line 2 spans from line 3 to 4. It is sequentially followed by the loop on line 5 because they are on the same level. Note that the software checks the consistency of the level numbering to a certain extent. E.g., it does not allow loop levels to increase by more than one from one line to the next

- **Count** – defines the number of loops. Use 0 or Inf to define an infinite loop
- **Segm#** – the waveform segment that you would like to generate
- **Mrk.Offset** – specifies the marker position in units of samples relative to the beginning of the segment. A value of 0 will generate a marker coincident with the segment and 100 means 100 samples later. If the Mrk.Offset field is empty, no marker will be generated
- **Mrk.Width** – defines the pulse width of the marker signal in units of samples. It is only necessary to define the Mrk.Width along with the first non-empty Mrk.Offset because all marker widths have to be the same
- **Cond.Jump** – can contain the row number of another segment. In case of a BREAK command or external event input, the sequencer jumps to that segment. If the Cond.Jump field is empty, no conditional jump is configured for this block

The buttons on the right-hand side of the panel have the following functionality:

- **List all segments** – display a list of all defined waveform segments along with their length
- **Delete all segments** – deletes all waveform segments
- **Reset M8070B** – sends a *RST to the M8070B software. This should not be necessary, but it can help to have a clean starting point
- **Channel selection** – selects the channel(s) to which this sequence is applied. If you want to have independent sequences per channel, it is easiest to open multiple **Sequencer Setup** windows and have them side-by-side by clicking on the **Sequencer Setup** button in the IQtools main window again
- **Break** – simulates a BREAK event to force a conditional jump
- **Stop** – stops the signal generation in the AWG
- **Run (no download)** – starts the AWG without downloading a sequence description
- **Download and Run** – downloads the contents of the **Sequence Table** and starts the AWG

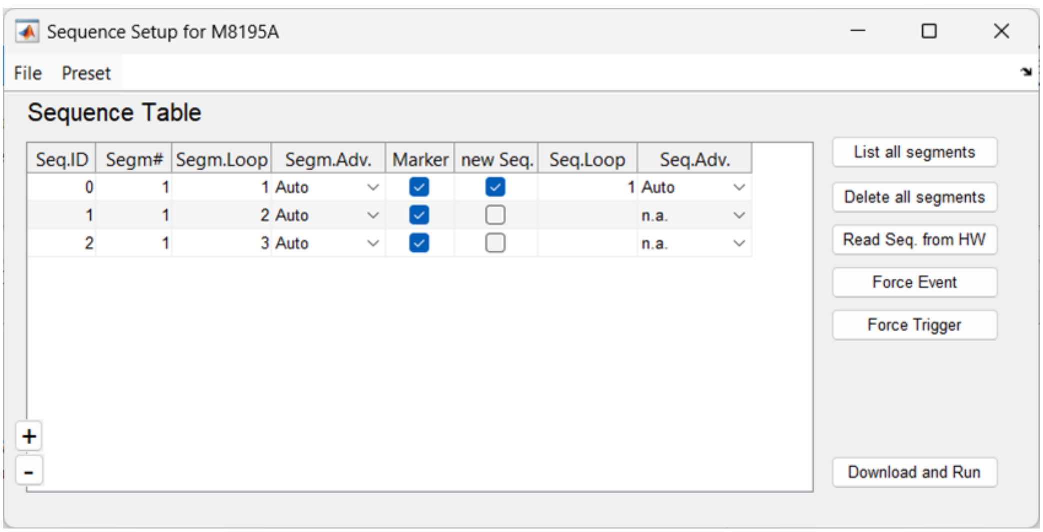
Menu Functions

- **File** -> **Load Settings**, **File** -> **Save Settings** – loads/saves the content of the **Sequence Table** to/from file
- **File** -> **Generate MATLAB code** – generates a piece of MATLAB code that does the same as clicking "Download and Run" with the currently defined sequence. The code can be directly executed in the code viewer window or saved to a file and used as a basis for programmatically controlling the sequencer from your own MATLAB script
- **File** -> **Generate XML** – generates the XML description of the sequence that is used to program the sequence into the M8070B software
- **Preset** – contains a few example sequences

Sequencer Setup for M8195A

The sequencer in the M8195A supports two levels of nested loops, but unlike the M8190A, it has a common sequencer for all channels. It has marker control as well as sophisticated control of how/when the sequences advances from one segment to the next. The inner loop level is called "segment" loop (i.e. repeating a certain waveform segment a certain number of times). The second loop level is called "sequence", i.e. repeating a series of segment loops. The sequencer will loop whatever is described in the **Sequence Table** infinitely.

Since the M8195A does not have digital upconversion engine built-in, it does not have the **Action Table**, **Frequency Table** and **Amplitude Table**, like the M8190A.



You can insert and delete rows from the table with the "+" and "-" buttons on the bottom left of the **Sequence Table**. In each row of the **Sequence Table**, you can specify the following:

- **Segm#** – the waveform segment that you would like to generate
- **Segm.Loop** – the number of times this waveform segment will be repeated
- **Segm.Adv**. – under which condition the sequencer will advance to the next segment

- **Marker** – whether the marker signal that has been defined for this segment will be generated
- **new Seq**. – when checked, marks the beginning of a new sequence
- **Seq.Loop** – the number of times the current sequence will be repeated
- **Seq.Adv**. – under which condition the sequencer will advance to the next sequence

The **Segment Advance** and **Sequence Advance** fields provides the following options:

- **Auto** – the waveform segment/sequence will be looped for the programmed number of times. Then, the sequencer will automatically proceed to the next table entry.
- **Conditional** – the segment/sequence will be looped until an external signal is applied to the ‘Event’ input (or SCPI command is sent to the instrument to simulate such an event). A segment will always be completed -independent on when the Event signal is asserted.
- **Repeat** – the waveform segment/sequence will be looped for the programmed number of times and then the output is paused as the last sample value. The sequence proceeds once the ‘Event’ input is asserted.
- **Stepped** – similar to ‘Repeat’, except that processing is paused after each loop.

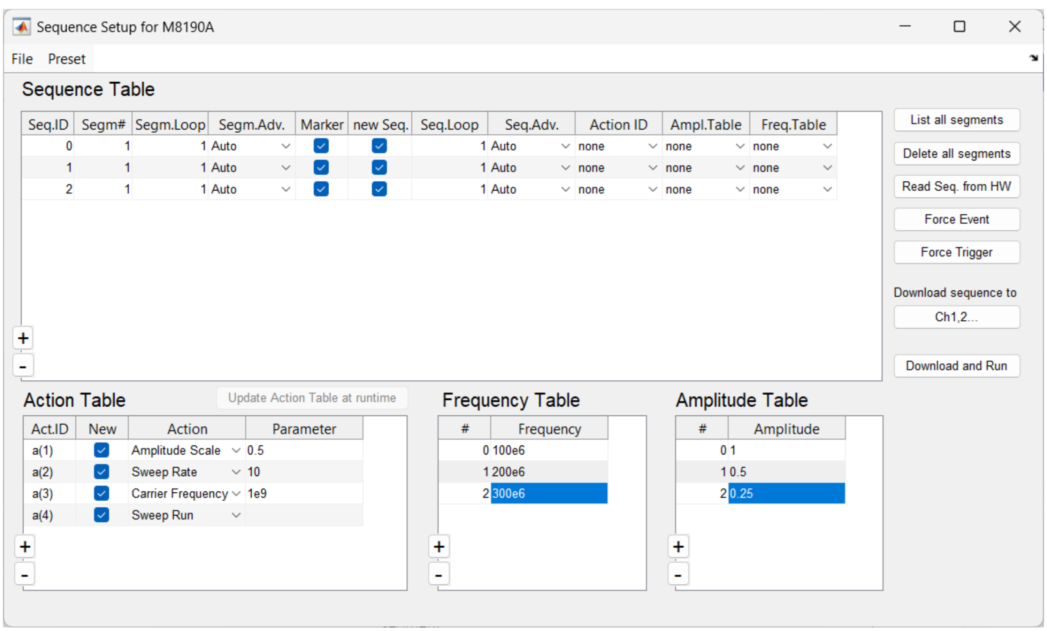
The buttons on the right-hand side of the panel have the following functionality:

- **List all segments** – display a list of all defined waveform segments
- **Delete all segments** – deletes all waveform segments
- **Read Seq. from HW** – uploads the sequence that is stored in the module into the sequencer window
- **Force Event** – simulates the assertion of the external "Event" Input
- **Force Trigger** – simulates the assertion of the external "Trigger" Input
- **Download and Run** – downloads the contents of the **Sequence Table** and starts the AWG

Sequencer Setup for M8190A

The sequencer in the M8195A supports two levels of nested loops, independent sequences per channel, marker control as well as sophisticated control of how/when the sequences advances from one segment to the next. The inner loop level is called "segment" loop (i.e. repeating a certain waveform segment a certain number of times). The second loop level is called "sequence", i.e. repeating a series of segment loops.

In Digital Upconversion (DUC) mode, the sequencer additionally supports **Frequency Tables** and **Amplitude Table** that allow hardware-controlled change of frequency and amplitude e.g. for frequency-hopping signals as well as sophisticated "actions", such as setting the frequency, amplitude and even starting and stopping a (hardware-controlled) extremely accurate frequency sweeps under the control of the sequencer.



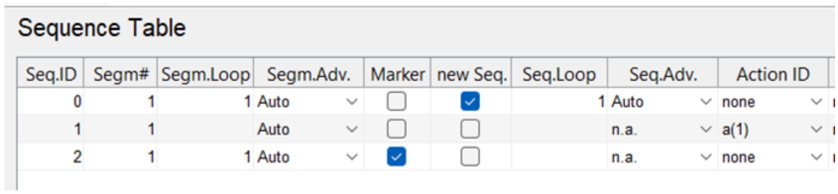
You can insert and delete rows from the table with the "+" and "-" buttons on the bottom left of the **Sequence Table**. In each row of the **Sequence Table**, you can specify the following:

- **Segm#** – the waveform segment that you would like to generate
- **Segm.Loop** – the number of times this waveform segment will be repeated
- **Segm.Adv**. – under which condition the sequencer will advance to the next segment
- **Marker** – whether the marker signal that has been defined for this segment will be generated
- **new Seq**. – when checked, marks the beginning of a new sequence
- **Seq.Loop** – the number of times the current sequence will be repeated
- **Seq.Adv.** – under which condition the sequencer will advance to the next sequence
- **Action ID** – can point to one of the "Act.ID"s in the **Action Table**
- **Ampl.Table** – the number of times the current sequence will be repeated
- **Freq.Table** – the number of times the current sequence will be repeated

The **Segment Advance** and **Sequence Advance** fields provides the following options:

- **Auto** – the waveform segment/sequence will be looped for the programmed number of times. Then, the sequencer will automatically proceed to the next table entry.
- **Conditional** – the segment/sequence will be looped until an external signal is applied to the ‘Event’ input (or SCPI command is sent to the instrument to simulate such an event). A segment will always be completed -independent on when the Event signal is asserted.
- **Repeat** – the waveform segment/sequence will be looped for the programmed number of times and then the output is paused as the last sample value. The sequence proceeds once the ‘Event’ input is asserted.
- **Stepped** – similar to ‘Repeat’, except that processing is paused after each loop.

In the **Action Table**, you can define groups of one or more individual actions. The "New"checkbox marks the beginning of a new group of actions. Action groups are automatically labelled with a(1), a(2), a(3) and so on in the "Act.ID" column. These Action IDs can be referred to from the main **Sequence Table**. When an Action ID is selected in the **Sequence Table**, it will be downloaded to the hardware during that waveform segment. However, it will only be triggered on the next segment that has the Marker enabled. In the following example, action ID a(1) is defined in Seq.ID 1 and triggered in Seq. ID 2.



Sequence Table

Seq.ID	Segm#	Segm.Loop	Segm.Adv.	Marker	new Seq.	Seq.Loop	Seq.Adv.	Action ID
0	1	1	Auto	☐	☑	1	Auto	none
1	1		Auto	☐	☐		n.a.	a(1)
2	1	1	Auto	☑	☐		n.a.	none

The available actions are listed in the following table, along with their parameter:

- **Phase Offset** – sets the phase offset register to a particular value. This offset is added to the phase accumulator. The same offset value is used until it is set to a different value. The parameter is specified in units of degrees
- **Phase Bump** – adds or subtracts a value from the phase accumulator. The parameter is specified in units of degrees
- **Phase Reset** – sets the phase accumulator to the specified parameter value. The parameter is specified in units of degrees
- **Sweep Rate** – sets the rate of frequency change in preparation for a Sweep Run action. The parameter is specified in units of Hertz per second. Values can be positive for an upward frequency sweep or negative for a downward frequency sweep
- **Sweep Run** – starts a frequency sweep. You should previously set the carrier frequency and sweep rate
- **Sweep Hold** – stops a frequency sweep without changing the carrier frequency
- **Carrier Frequency** – sets the DUC carrier frequency. Parameter is in units of Hertz
- **Amplitude Scale** – sets the DAC scaling to a value between 0 and 1

The **Frequency Table** provides a way to step through a predefined sequence of carrier frequencies, e.g. simulating a frequency hopping signal. The list of carrier frequency values is defined in the **Frequency Table**. In the main **Sequence Table**, the “Freq. Table” column specifies when the next frequency value is selected from this list:

- init – resets the pointer to the beginning of the **Frequency Table**.
- next – increments the pointer to the next frequency value.
- none – keeps the pointer unchanged.

The **Amplitude Table** works the same way.

The buttons on the right-**a**hand side of the panel have the following functionality:

- **List all segments** – display a list of all defined waveform segments
- **Delete all segments** – deletes all waveform segments
- **Read Seq. from HW** – uploads the sequence that is stored in the module into the sequencer window
- **Force Event** – simulates the assertion of the external "Event" Input
- **Force Trigger** – simulates the assertion of the external "Trigger" Input
- **Download sequence to** – selects the channel(s) into which the current sequence is downloaded
- **Download and Run** – downloads the contents of the **Sequence Table** and starts the AWG

Using the Sequencer Programmatically

The sequencer functionality can also be accessed programmatically from within a user’s MATLAB script by calling the **`iqseq()`** function. The easiest way to start with programmatic control of the sequencer is to set up a sequence in the **Sequencer Setup** window then click on **File** -> **Generate MATLAB code**. You can run the code directly from the code viewer or copy it into a **`.m`** file and make the desired modifications.

For a detailed description of how to use `iqseq.m`, refer to the comments in the header of **`iqseq()`**.

In addition, the sequencer demo that comes with IQTools (**`seqtest1.m`**) contains an example implementation.

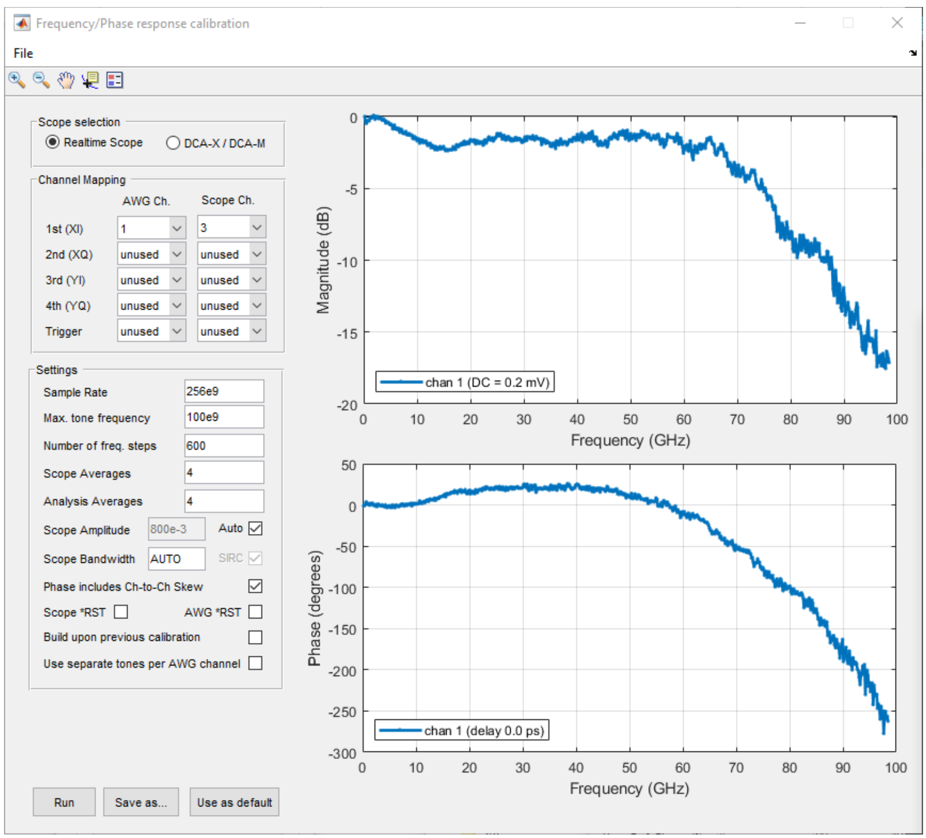
You can also access the sequencer directly via SCPI commands. For more information, refer to the programming guide of the respective AWG model.

In-System Calibration

This feature allows you to calibrate the instrument for accurate waveform generation/acquisition by performing various analyses on the instrument, plotting various graphs, and extracting the data to display calibrated data.

- To use the In-System Calibration utility, click **In-System Calibration** in the IQTools main window or the Serial Data window. You can also use File -> In-system calibration in the main window.

  The following figure shows the **Frequency/Phase response calibration** window.



This window displays how the data is calibrated as per channel mapping and other connections. Whenever the user opts for in-system calibration, these calibrations will automatically be performed, provided the required hardware connection is available.

To achieve an optimal signal performance out of the AWG, it is recommended to run an ‘in-system calibration’. The in-system calibration determines the frequency- and phase response of the AWG plus the cables and adapters that are between AWG and scope. It also determines the exact skew over frequency between I and Q, which is important for a valid complex modulation signal.

In the **Frequency/Phase response calibration** screen, you can select the desired type of scope and define how you have connected the AWG channels with the scope. If you plan to calibrate only a single channel, you can leave all the other parameters unchanged and click **Run**.

Once the calibration executes successfully, you will see the magnitude and phase response graphs on the right side of the screen.

Here is a description of the controls in the “Settings” pane. In the following section, the term “*user signal*” refers to the waveform that the user plans to play back once the in-system calibration has been completed.

- **Sample Rate**: The sample rate of that will be used during the calibration process. To achieve optimal results, it is recommended that sample rate used for the in-system calibration is the same that will be used for the *user signal*.
- **Max tone frequency**: This is the frequency up to which the in-system calibration is performed. This frequency must be equal or greater than the highest frequency component in the user signal. On the other hand, the max tone frequency should be selected such that the magnitude of the system does not fall more than approx. 20 dB below the magnitude at low frequencies. This will cause the measurement tones to be “buried” in the noise, which does not allow the algorithm to measure the magnitude or phase of the respective tones. If this happens, you will see “Measurement outlier” errors. If this happens, reduce the max tone frequency.
- **Number of frequency steps**: This value determines the resolution of the measurement on the frequency axis. For example, if the max tone frequency is 12 GHz and the number of frequency steps is 600, the tone spacing will be 12 GHz/600 = 20 MHz. The number of tones is a compromise between fine resolution (= large number of tones) and measurement accuracy (= small number of tones). Numbers between 300 and 1000 are usually a good choice.

- **Scope Averages**: Number of averages in the scope acquisition. This number is only used in case a triggered measurement is made. Otherwise, this setting is ignored. Triggered measurements are no longer recommended. For the Real-time scope, set the "Trigger" in the Channel Mapping pane to "unused". For the DCA, set it to "PTB+FP Trigger".
- **Analysis Averages**: This determines how many times the overall measurement is repeated. The final result is the average of the individual magnitude resp. phase responses. Increasing the number of Analysis averages reduces statistical measurement errors (that is, noise), but it increases the amount of time it takes to run the calibration. Using four averages is a good compromise.
- **Scope amplitude/Autoscale**: If the Autoscale check box is checked, the scope performs an autoscale before making a measurement. Otherwise, you have to provide the full range voltage in the scope amplitude field.
- **Scope BW**: Sets the scope bandwidth to a particular value or "MAX" resp. "AUTO" for the maximum (resp. default) value.
- **SIRC**: Selects the type SIRC filter for the DCA. In general, "Flat" is a good choice for the calibration.
- **Scope *RST/AWG *RST**: Sends a *RST command to the Scope resp. AWG before each measurement. Not recommended.
- **Build upon previous calibration**: If checked, the algorithm applies the previously captured correction to the test signal when running an in-system calibration. In theory, you should see a (nearly) straight line in both magnitude and phase. You can use this for verification, but it generally does not improve the signal quality if multiple calibrations with this check box set are run in sequence.
- **Use separate tones per AWG channel**: This check box must be checked when the signal from two (or more) AWG channels are observed on the SAME scope channel. An example would be if you have AWG Ch1+ and Ch2- connected to the two inputs of a balun and observe the output of the balun on the scope. Another example is measuring through an optical link where the polarization is potentially scrambled on the way.

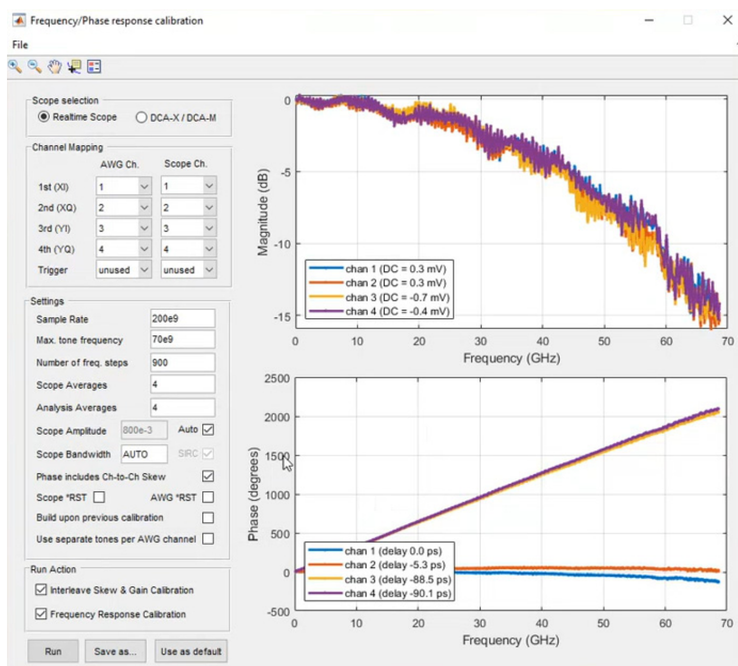
M8199A/B Specific Calibration

M8199A User Calibration In-System Calibration

This feature allows you to store the calibrated value in the user calibration area using the cal table on the module itself.

1. Setup the M8199A multi-module configuration. For more information, see “Instrument Configuration” on page 64.
2. Click **In-System Calibration** under the **Sequencer** pane.

   The following figure displays the **Frequency/Phase response calibration** parameters.



   This window displays how the data is calibrated as per channel mapping and other connections. Whenever the user opts for in-system calibration, these calibrations will automatically be performed, provided the required hardware connection is available.

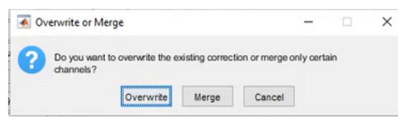
3 Configure the following parameters as required:
- Scope Selection
- Channel Mapping
- Settings
- Run Action

4 Click **Run**.

The calibration between the AWG channels and Scope channels will be performed.

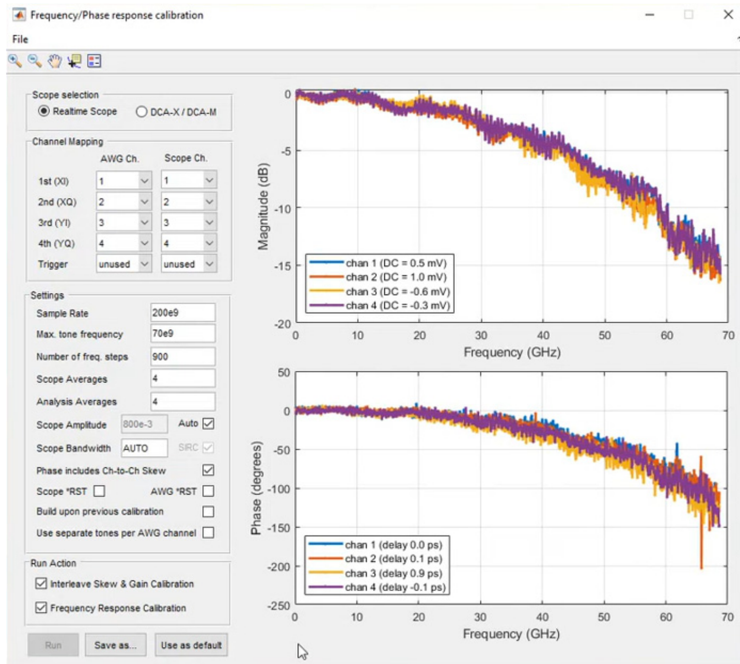
5 To de-skew the scope channels, click **Run** again. This will consider channel 1 as the benchmark and performs signal de-skew between channels (1-2), channels (1-3), and channels (1-4).

The **Overwrite or Merge** dialog box appears. The following options are available:
- Overwrite: To store the calibrated value in the user cal table.
- Merge: To merge only certain channels.
- Cancel: To cancel the calibration.



6 Click **Overwrite** to store the calibrated value in the user cal table.

Once the calibration executes successfully, you will see the magnitude and phase response graphs on the right side of the screen.



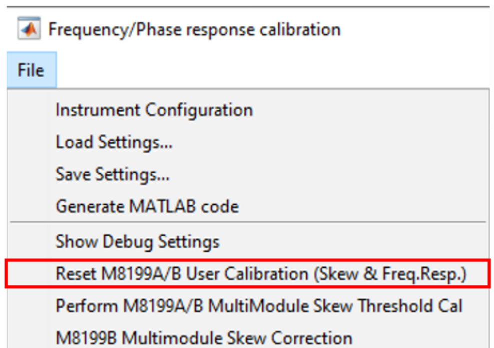
Frequency/Phase response calibration
File
Scope selection
Realtime Scope
DCA-X / DCA-M
Channel Mapping
AWG Ch.
Scope Ch.
1st (XI)
2nd (XQ)
3rd (YI)
4th (YQ)
Trigger
unused
Settings
Sample Rate 200e9
Max. tone frequency 70e9
Number of freq. steps 900
Scope Averages 4
Analysis Averages 4
Scope Amplitude 800e-3 Auto
Scope Bandwidth AUTO SIRC
Phase includes Ch-to-Ch Skew
Scope *RST
AWG *RST
Build upon previous calibration
Use separate tones per AWG channel
Run Action
Interleave Skew & Gain Calibration
Frequency Response Calibration
Run
Save as...
Use as default
Magnitude (dB)
chan 1 (DC = 0.5 mV)
chan 2 (DC = 1.0 mV)
chan 3 (DC = -0.6 mV)
chan 4 (DC = -0.3 mV)
Frequency (GHz)
Phase (degrees)
chan 1 (delay 0.0 ps)
chan 2 (delay 0.1 ps)
chan 3 (delay 0.9 ps)
chan 4 (delay -0.1 ps)
Frequency (GHz)

Reset M8199A User Calibration

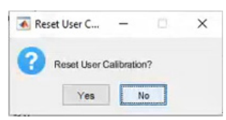
This feature allows you to reset the M8199A user calibration. This will clear all the older values from the cal table.

1 Click **In-System Calibration** under the **Sequencer** pane.

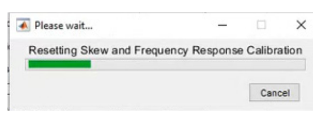
2 Go to **File** > **Reset M8199A User Calibration**.



3 On the **Reset User Calibration?** dialog box, click **Yes**.



The reset process starts.



4 Click **Run**.

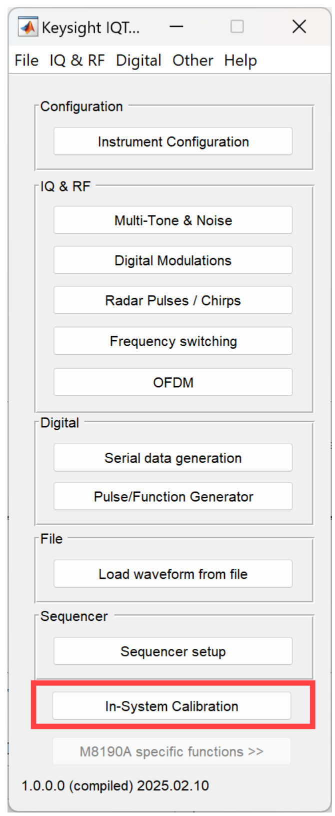
Once the reset process is complete, you will see the magnitude and phase response graphs on the right side of the screen.

M8199A Multi-Module Skew Calibration

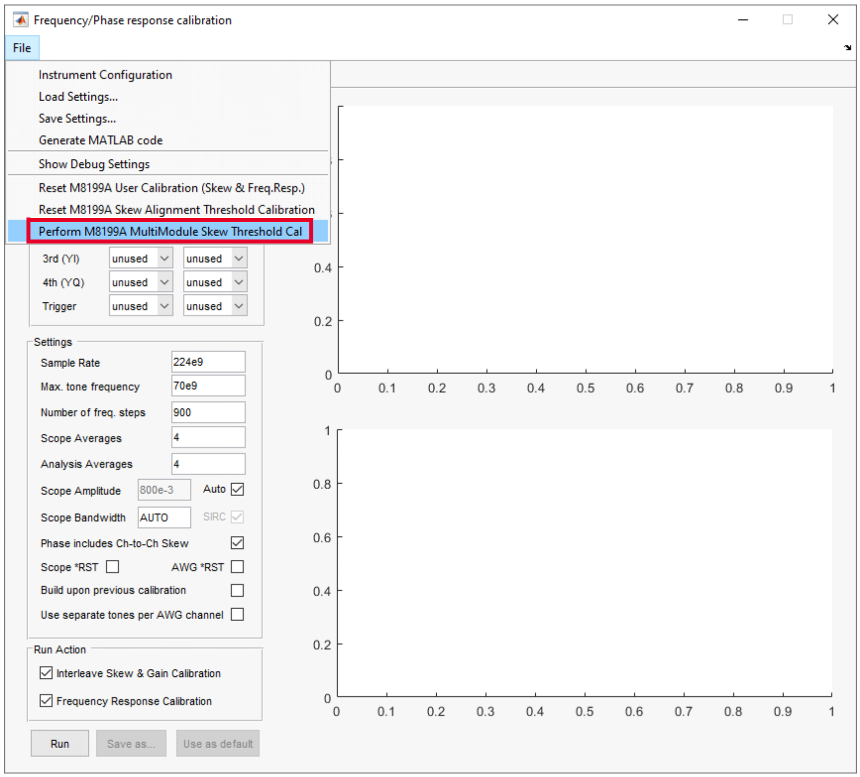
Multi-module skew calibration consists of two steps: The **Multi-Module Skew Threshold Cal** makes sure that the skew between modules is repeatable while the **Frequency/Phase Response Cal** measures and compensates any remaining channel-to-channel skew. The results of these measurements are stored in the non-volatile user calibration area on the module.

1 Setup the M8199A multi-module configuration. For more information, see "Instrument Configuration" on page 64.

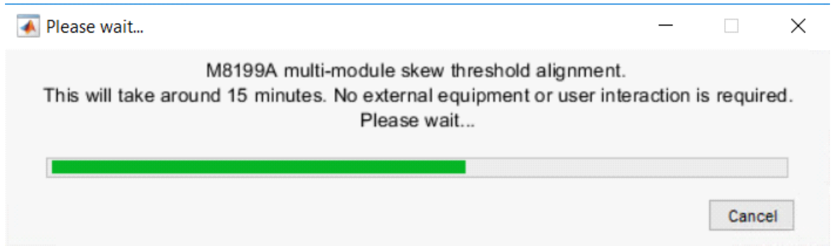
2 Click **In-System Calibration**, which can be found below the **Sequencer** pane.



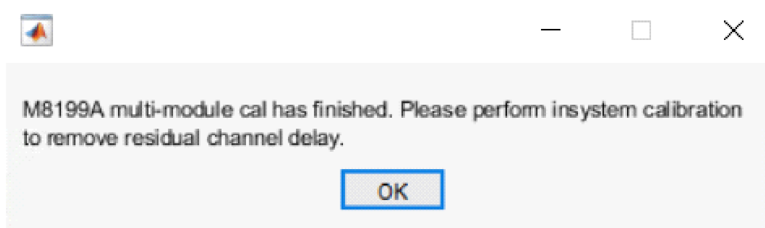
3 On the main menu of the **Frequency/Phase response calibration** window that appears, click **File** > **Perform M8199A Multi-Module Skew Threshold Cal**.



The following progress window is displayed when you click the option to perform the **M8199A Multi-Module Skew Threshold Cal**. Note that this step takes approximately 15 minutes to complete.



The following dialog box is displayed after the multi-module skew threshold alignment is complete.

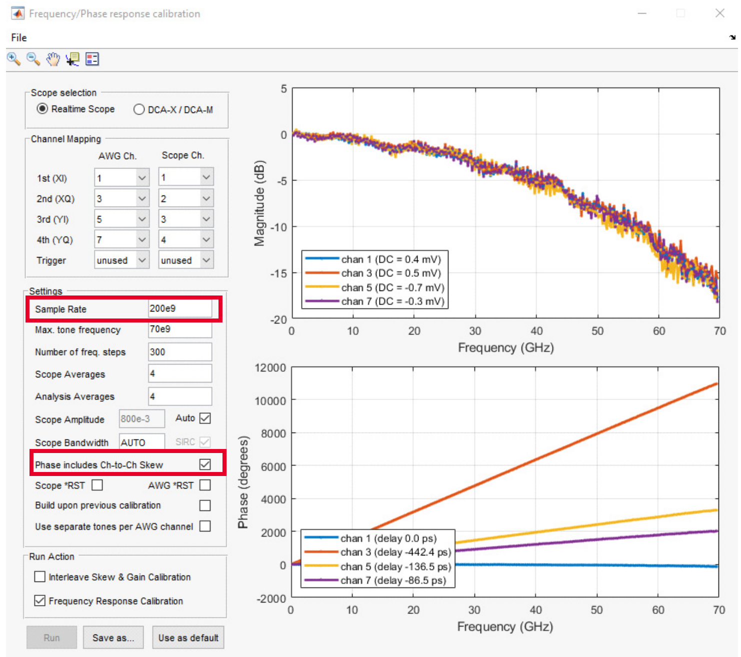


4 Click **OK** to exit and return to the previous window.

5 Perform the In-System calibration to compensate Channel-to-Channel skew using at least three different sample rates, that is,

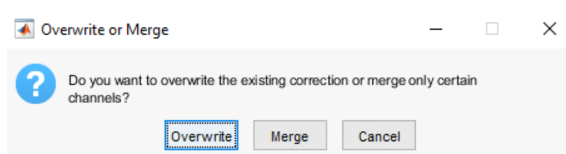
- 200 GSa/s
- 228 GSa/s
- 256 GSa/s

The **Frequency/Phase response calibration** window for sample rate 200 GSa/s is shown below:



NOTE The number of tones (Number of freq steps) must be reduced to ‘300’, otherwise errors pertaining to “phase outliers” could occur during measurement. Also, the “Phase includes Ch-to-Ch Skew” option must be enabled, otherwise the user cal table will not be updated correctly.

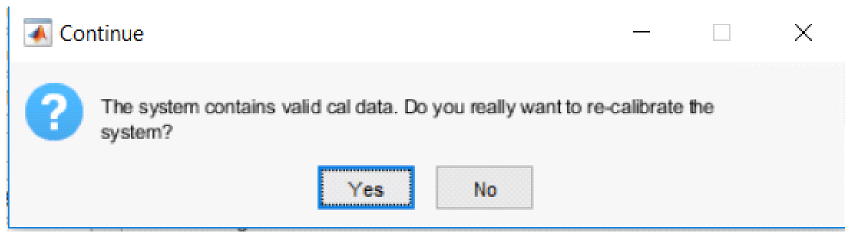
6 Click **Overwrite** on the following prompt at the end of the calibration with each sample rate.



Here are some important points to note:

- The M8199A multi-module skew threshold calibration does not require any external measurement equipment.

- It is mandatory to use the official Keysight Clock Cables (M8199A-810) and Sync Cables (M8199A-811).
- The calibration routine sweeps through the sample rate range in steps of 500 MSa/s (or ILV 1 GSa/s) and can be performed in both (non-interleaving and interleaving) modes.
- One calibration cycle requires at least 15 minutes.
- Multimodule cal data is valid, as long as the module order and the cabling do not change.
- You will be informed if valid Calibration data are available. If not, you have the choice to perform the calibration again. The following prompt appears in such cases.



- In-system calibration requires an N1000A+N1046A or a UXR with at least 70 GHz of bandwidth.
- When more than four channels have to be aligned, you must run the IQTools In-System calibration more than once; (IQTools) Channel 1 is the reference channel and must be used in each run.

For instance, if eight Channels have to be aligned,

1 insystem cal #1: ch1, ch2, ch3, ch4
2 insystem cal #2: ch1, ch5, ch6, ch7
3 insystem cal #3: ch1, ch8

M8199B Multi-Module Skew Calibration

Overview

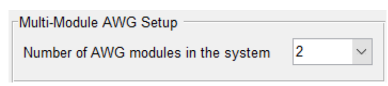
While individual M8199B AWG modules are pre-calibrated from the factory, in order to ensure phase stability between multiple M8199B modules, you must perform a user calibration, which involves two steps.

1 A skew threshold calibration must be performed to avoid UI skew errors when the sample rate has been changed. This calibration does not require external equipment, but takes several minutes.This calibration is valid, as long as any of the system components (such as, modules, clock, sync cables, and so on) remain unchanged.

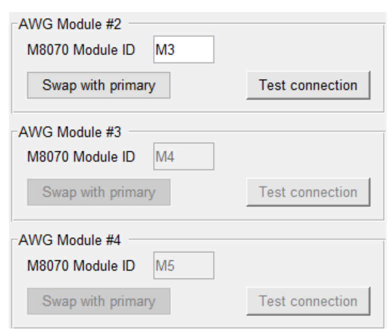
2 Once the previous steps is complete, the skew between the modules is stable, but not zero. Therefore, a second calibration step can be performed, which measures the skew between modules and writes it into a user cal table.

Step 1: Multi-module skew threshold calibration

1 On the **Instrument Configuration** window of the IQTools GUI, in the **Multi-Module AWG Setup** area, select at least ‘2’ modules.

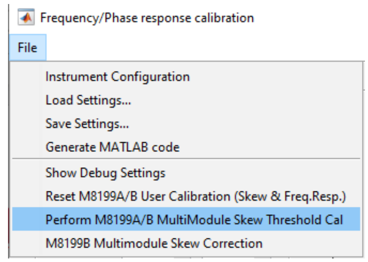


2 Provide the correct module IDs.



3 Launch the **In-System Calibration** window.

4 Click **File** > **Perform M8199A/B MultiModule Skew Threshold Cal**.



Note that the IQTools utility automatically checks and prompts you in case a skew threshold cal was already performed with the current module configuration. For example, when the clock/sync cable between M8199B and M8008A has been changed, a new calibration is required. In such cases, the IQTools utility starts the Multi-Module Skew Threshold calibration process.

While the calibration is being performed, a progress window is displayed.

5 Wait for the calibration to finish.

Step 2: M8199B Multimodule Skew Correction

Channel mapping configuration

As mentioned earlier, after the skew threshold calibration is performed, the skew between two or more M8199B modules is stable, but not zero and is dependent on the AWG sample rate.

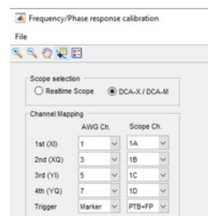
IQTools utility helps you with the skew correction also. To perform this step, you require an oscilloscope (real-time/sampling oscilloscope).

To perform the Multi-Module Skew Correction, 1 Connect channel 1 of each module (either normal or complement, not mixed) to a channel on the oscilloscope.

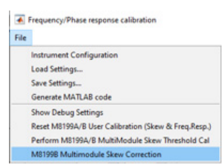
NOTE

Make sure that matched cables are used to perform connections between the M8199B modules and the oscilloscope. Any skew, which is not coming from the AWG, will be measured and corrected/calibrated as well.

2 Select the corresponding mapping on the **In-System Calibration** window.



3 Click **File** > **M8199B Multimodule Skew Correction**.



The skew is then measured between the channels. At the end of the calibration process, the measured values are written in a user cal table on each module. Therefore, even after a power cycle of the chassis, the skew values are retained.

Correction Management

Concept of Operation

Each of the waveform generation windows has an “Apply correction” check box. This check box determines whether the calculated waveform will be pre-distorted to compensate for the system’s frequency/phase response. The details on which correction(s) are applied are visualized and controlled in the correction management window.

IQTools uses two types of corrections: **complex** and **per-channel** correction. Complex corrections are applied to a complex-valued waveform before channel mapping while per-channel corrections deal with real-valued waveforms after channel mapping. In both cases, the correction information is complex-valued (that is, contains frequency and phase correction).

Complex corrections are derived from the VSA equalizer (see “Digital Modulations” on page 77 for more details), while **per-channel** corrections are derived through in-system calibration by capturing time-domain data using an oscilloscope or DCA (see “In-System Calibration” on page 131) and through loading external S-parameter files.

IQTools implicitly saves the “current” correction parameters in a file (*C:\Users\<username>\AppData\Local\Keysight\iqtools\ampCorr.mat*) to keep them persistent across re-start of MATLAB and the IQTools application. In addition, corrections can be saved and loaded from a file – see Export/Import below.

When “Apply corrections” is enabled, both complex AND per-channel corrections are applied sequentially. In a first step, the complex-valued correction is applied to the complex-valued waveform:

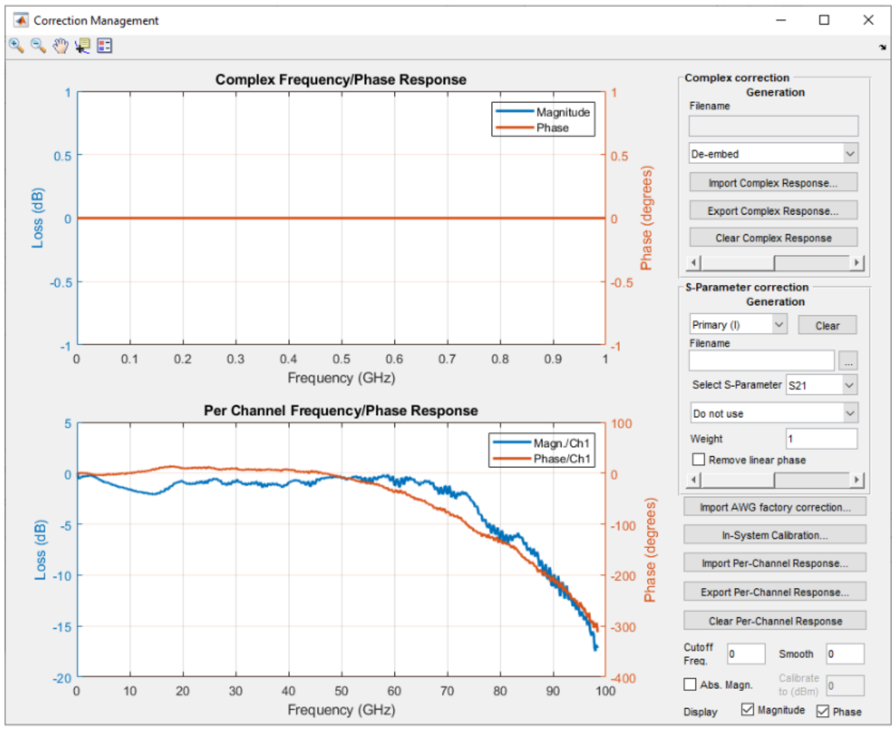
$$sig' = IFFT(FFT(sig) * complexCorrection)$$

Next, the real and imaginary parts of the waveform are mapped to AWG channels (as defined by the channel mapping) and the per-channel corrections for the specific channel is applied:

$$\begin{aligned} sig\_I &= IFFT(FFT(real(sig') * perChannelCorrection) \\ sig\_Q &= IFFT(FFT(imag(sig') * perChannelCorrection) \end{aligned}$$

Correction Management Window

The correction management window can be accessed from the main window under **File** > **Show Corrections**. Alternatively, most of the waveform generation windows contain a “Show Corrections” button.



The following plots and controls are available on the correction management window:

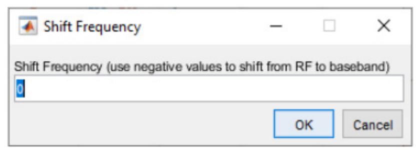
- **Complex Frequency/Phase Response plot** visualizes the current complex correction. Magnitude (blue) and phase (red) is shown as a separate graph.

  A straight horizontal line at 0 dB means no complex correction is being performed.

- **Per Channel Frequency/Phase Response plot** visualizes the current per channel correction. Magnitude and phase per channel are shown as separate graphs. The two check boxes on the bottom right of the window can be used to turn off magnitude or turn off phase for better visibility.

Complex correction panel

- **Embed/De-embed/Do-not-use** selects whether the complex correction itself or the inverse of the correction is multiplied with the original waveform.
  - **Import Complex Response** loads the complex response from a file. Supported formats are .MAT, .CSV (VSA style) or .CSV (Signal Optimizer style). After selecting a file, the following dialog opens up:



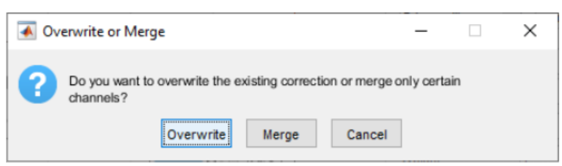
By entering a non-zero value, the frequencies in the correction file can be shift right (with positive) or left (with negative) frequency values. A shift is necessary if VSA captures a signal that has been externally up-converted, but the correction should be performed at baseband. In this case, a negative value equal to the carrier frequency should be entered. If the AWG generates a direct RF signal, the correction signal must not be shifted.

- **Export Complex Response** saves the complex response in a file. Supported formats are .MAT and .CSV (compatible with the "VSA style" format in Import Complex Response).
- **Clear complex response** clears the "current" complex response. A straight horizontal line will appear in the complex response plot to indicate that there is not complex response available.

S-parameter panel

The S-parameter panel "belongs" to the per-channel corrections. The per-channel plot reflects the per channel corrections that were imported from a file or acquired through in-system calibration PLUS the S-parameter corrections. This can be useful in order to perform some S-parameter "math" or to apply multiple S-parameter files.

- **Primary/Secondary/3rd/4th** pull-down menu switches between up to 4 different S-parameter files. If only the Primary file is specified, it will be used for all channels. If a primary and secondary are specified, the primary will be used for channel 1 and the secondary for all remaining channels and so on.
- **Clear** button clears all S-parameter file assignments.
- **Selected S-parameter** defines which column in the S-parameter file is used for correction. Usually this is S21 (insertion loss).
- **Embed/De-embed/Do-not-use** determines if the content of the S-parameter file is embedded (that is, “added”), de-embedded (that is, “removed”) from the waveform or not used at all. By default, de-embed is the correct choice.
- **Weight** defines how many times the S-parameters are embedded/de-embedded. For example, if the S-parameter file contains the response of a 1 m cable, a weight of 2 can be used to embed/de-embed a 2 meter cable. Weight can be fractional.
- **Remove linear phase** takes out a constant delay (= linear phase) from the S-parameter file before applying it.
- **Import AWG factory correction** recalls the correction coefficients that were stored in the AWG as part of the manufacturing process. Only M8195A, M8196A, M8194A, M8198A, M8199A and M8199B have built-in factory corrections stored in the module.
- **In-System Calibration** opens the in-system calibration window (see “In-System Calibration” on page 131).
- **Import per-channel response** loads the per-channel response from a file. Once the type of file and filename are selected, the user can choose to “Overwrite”, “Merge” or “Cancel”.



  - “Overwrite” means that the current corrections are deleted and completely replaced by the newly loaded ones.
  - “Merge” means that corrections for existing channels remains intact and the newly loaded corrections will be used for other channels. (For example, you already have corrections for channel 1 and want to load corrections for channel 2 without losing those for channel 1.

Supported file formats are:

- **.S2P/S4P/S6P/...** – S-parameter file. After selecting a file, you can choose which S-parameter columns should be used (for example, 2 1 4 3 means S21 and S43) and to which AWG channel(s) the file should be assigned (for example, 2 3 means AWG channel 2 and 3).
- **.MAT** – MATLAB file – compatible with the .MAT format in “Export per-channel response”. Once the filename is selected, the user can define to which AWG channels the file is assigned.
- **.CSV** – This file format corresponds to the .CSV format in “Export per-channel response”. Once the filename is selected, the user can define to which AWG channels the file is assigned.

▪ **Export per-channel correction** saves the per-channel information in a file. Supported file formats are:

- **Touchstone SnP** – S-parameter file with N = 2*number of channels. Saves the correction for all channels in a single S-parameter file. Columns are S21, S43, S65, and so on.
- **Touchstone S2P** – individual S-parameter files, one per channel.
- **.MAT** – MATLAB format – compatible with the .MAT format in “Export per-channel response”.
- **.CSV** – compatible with .CSV format in “Import per-channel response”.
- **.CSV (VSA style)** – .CSV file compatible with VSA.
- **.CAL (VSA user correction)** - can be loaded as a user correction in VSA.

▪ **Clear per-channel response** clears data from the per-channel response. If corrections for multiple channels is available, a dialog pops up that allows you to pick which channel(s) to clear. (For example, 1 3 clears only corrections for channels 1 and 3).

▪ **Cut-off frequency** can be used to reduce the upper frequency limit up to which corrections are applied without re-running in-system calibration. For example, if the per-channel response plot contains corrections up to 100 GHz, but corrections shall only be applied up to 70 GHz, the cut-off frequency can be set to 70G.

Note that the remaining data (from 70 to 100 GHz) is not removed, it is just not used and not displayed in the per-channel response graph. Once the Cut-off frequency field is set to 0, the complete frequency response information is available again.

- **Smoothing** applies a smoothing filter to the magnitude and phase response curves. The smoothing value indicates across how many frequency points the smoothing should take place. For example, smoothing = 20 shall smoothen across 20 points. Set **smoothing** to zero to disable this function.

M8190A Specific Functions

It allows you to control some more M8190A specific functions, such as, CATV Signal Generation, Sequencer Demo (Time Domain), Radar Demo (Moving Target), 4-Channel Synchronization, and so on.

Using the M8190A in Digital Up-Conversion Mode

The IQTools utilities (except iserial) can also be used to generate baseband signals for the M8190A operating in **Digital Up-conversion** (DUC) mode. In order to use DUC mode, use the configuration window and select one of the DUC mode (x3, x12, x24 or x48) in the **Instrument Model** pop-up menu. Optionally, you can also select the carrier frequency in the same window. After you click **OK**, open any of the IQTools utilities and define the baseband waveform.

Notice that the **Download** pop-up menu now shows **RF to channel 1** and **RF to channel 2** as the possible selections.

Working with Two M8190A Modules Simultaneously

IQTools supports a 4-channel setup that consists of two M8190A modules with an optional M8192A synchronization module.

Working Without the M8192A SYNC Module

It is possible to synchronize two M8190A modules with the help of an oscilloscope down to approx one ps skew between each pair of channels. For more information, see "M8190A-Specific Utilities: 4-Channel Synchronization" on page 153.

Working With the M8192A SYNC Module

When working with the M8192A SYNC module, ensure the following:

- You have an M8190A firmware instance running for each of the two modules.
- You have an M8192A firmware/SFP running.
- You have both M8190A VISA addresses and the M8192A VISA address configured in the IQTools configuration window.

With these prerequisites, the **Download To** button in each of the utilities allows you to select to which of the four channels your real and imaginary part of the waveform will be downloaded. It is possible to load the same component to multiple channels.

M8190A-Specific Utilities: 4-Channel Synchronization

This utility allows you to demonstrate the synchronization of two M8190A modules (= 4 channels) either with or without an M8192A synchronization module. It is generally possible to synchronize more than two M8190A modules (up to 6 with the M8192A sync module), but the current implementation of IQTools only supports two.

To run four M8190A channels fully synchronous, follow these steps:

1 Start the M1890A firmware for the first and second module (you need one instance of the firmware for each module).

2 If you are using the M8192A module, launch the Soft Front Panel for the M8192A module (this includes the firmware). Find the VISA address of the M8192A (it is shown under Help > About).

3 Find out and note the IP address of the oscilloscope and add the oscilloscope to the Keysight IO Connection Expert. If LAN (TCPIP) does not work reliably, try to use USB to connect to the scope. Any DSO, DSA or MSO scope will work.

4 On the IQTools GUI, click **Configure Instrument**.

   a Select M8190A_12bit or M8190A_14bit mode. Digital upconversion mode are currently not supported in IQTools (although synchronization works in the same way)

   b Enter the VISA addresses for both AWG modules (just copy them from the firmware window – you might want to click the **Test connection** button to be sure that connection can be established.

   c Enter the VISA address of the scope (copy it from the IO Control, Connection Expert) – you might want to click the **Test connection** button to be sure that connection can be established.

   d Enter the VISA address of the M8192A if you are using a SYNC module. It is important that your hardware configuration and cabling matches the configuration that you set up in IQTools. If you are not using an M8192A module, make sure to clear the **Use M8192A** check box.

5 From the main screen of the IQTools MATLAB GUI, select > specific functions > 4-channel sync.

6 Make the cable connection described in the Connection Diagram.

7 Set the **AWG#2 Clk Source** to **ext sample clock** for best accuracy. (Make sure you have the Sample clocks connected according to the connection diagram). If you are using the M8192A SYNC module, the sample clock connection is part of the blue cable connections.

8 Make sure **Analog Outputs Ch1 used for de-skew ...** is selected.

9. Click **Automated De-skew**. This will synchronize the two first channels on both modules.

   If an error message pops up, look at the scope – it should display the rising edge of a square wave on channel 1 and 2. If one (or both) of the channels show no signal, double check the connection diagram. Also, make sure you using the correct output (DAC out vs. DC out).
10. The modules are now synchronized – you should see all 4 scope channels perfectly aligned on the oscilloscope screen.
11. Choose one of the waveforms in the drop-down menu and click the **Start** or **Stop** buttons to see how different waveform can be loaded without changing the skew between the modules.

For Customer Specific Waveform the Script must be modified:

1. Open the file multi_channel_sync.m for editing in MATLAB.
2. Go to line 202 and decide which item of the sample list you would like to customize.
3. The waveformID is linked to the point in the drop-down menu of the GUI. This gives you an idea which case needs to be modified.
4. The waveform vectors testSegment1 or testSegment2 have to be complex due to the IQ philosophy of the scripts.
5. testSegment1 is then downloaded to AWG#1 (primary).
6. testSegment2 is then downloaded to AWG#2 (secondary).
7. Once the modifications are done, save the .m-file and restart the GUI for 4-channel-sync.
8. Select the modified waveform in the drop-down menu and play it by pressing the buttons **Start** and **Stop**.

M8199B with M8159A

An M8199B with external multiplexer M8159A setup consists of two to four M8199B modules, one or two M8159A boxes plus corresponding cables, attenuators and balun.

To control M8199B/M8159A setup using the IQTools needs initial configuration and calibration. For more information, refer to "IQTools for M8199B with M8159A User Guide".

4 References

References

- Learn more about the Keysight M8190A arbitrary waveform generator (used in this document) at www.keysight.com/find/M8190A
- Learn more about the Keysight M8194A arbitrary waveform generator (used in this document) at www.keysight.com/find/M8194A
- Learn more about the Keysight M8195A arbitrary waveform generator (used in this document) at www.keysight.com/find/M8195A
- Learn more about the Keysight M8196A arbitrary waveform generator (used in this document) at www.keysight.com/find/M8196A
- Learn more about the Keysight M8198A arbitrary waveform generator (used in this document) at www.keysight.com/find/M8198A
- Learn more about the Keysight M8199A arbitrary waveform generator (used in this document) at www.keysight.com/find/M8199A
- Learn more about the Keysight M8199B arbitrary waveform generator (used in this document) at www.keysight.com/find/M8199B
- Learn more about the Keysight M8070B system software at www.keysight.com/find/M8070B
- Learn more about MATLAB software and ordering it directly from Keysight with the M8190A, other arbitrary waveform generators, and other instruments at: www.keysight.com/find/MATLAB
- Information on all Keysight signal and waveform generators can be found at www.keysight.com
- Additional information on using MATLAB with Keysight instruments is available at www.mathworks.com/keysight

KEYSIGHT
TECHNOLOGIES

Edition 7.0, November 2025
M8100-91B60

www.keysight.com