R&S SMW200A csv to XDW Python Script Instructions

Document Version 1.0

09/06/2024

Written by:

Luke Grantham

Application Engineer

Rohde & Schwarz USA

[luke.grantham@rsa.rohde-schwarz.com](mailto:luke.grantham@rsa.rohde-schwarz.com)

(619)930-6753

A blue and white background with arrows

Description automatically generated

Contents

[1 About 3](#_Toc176525782)

[1.1 Purpose 3](#_Toc176525783)

[1.2 Key Features 3](#_Toc176525784)

[2 Getting Started 3](#_Toc176525785)

[2.1 Required Components 3](#_Toc176525786)

[2.2 What’s Included 3](#_Toc176525787)

[2.3 Installation Instructions 4](#_Toc176525788)

[2.3.1 Pip 4](#_Toc176525789)

[2.3.2 Numpy 4](#_Toc176525790)

[2.3.3 Rsxdwstreaming 4](#_Toc176525791)

[2.3.4 Rsfkd 5](#_Toc176525792)

[2.4 Overview of Operation 5](#_Toc176525793)

[2.5 References 5](#_Toc176525794)

[3 CSV Structure 5](#_Toc176525795)

[4 Workflow 7](#_Toc176525796)

[4.1 Build a list of PDWs and TCDWs in .xlsx Format 7](#_Toc176525797)

[4.2 Save to .csv 7](#_Toc176525798)

[4.3 Run Python Script 7](#_Toc176525799)

[4.4 Transfer Script Output Files to the SMW200A 9](#_Toc176525800)

[4.5 Play the Files Using the Extended Sequencer 10](#_Toc176525801)

[5 Q&A 14](#_Toc176525802)

# About

## Purpose

This script was created to convert .csv formatted Pulse Descriptor Word (PDWs) and Timed Control Descriptor Words (TCDWs) to a format that is readable by Rohde & Schwarz vector signal generators.

Descriptor Words are simple, readable structures that describe a pulse. R&S Pulse Descriptor Words (PDW) can be used to generate pulsed signals in real-time or replay pre-calculated waveform segments. R&S Timed Control Descriptor Words (TCDW) can be used to change instrument RF frequency and/or level or re-arm the Extended Sequencer.

## Key Features

Customers provide a .csv with Descriptor Words with information describing a pulse. The script processes the .csv and generates the necessary files to output the pulses on the Rohde & Schwarz SMW200A Vector Signal Generator.

# Getting Started

## Required Components

On the PC running the script:

1. Python: Version 3.12.3 was used during verification.
2. rsxdwstreaming: the package file (wheel) for PDW Expert. Please reference installation instruction in Section 2.3 for more details.
3. numpy version 1.26.3. It most likely will work with any version previous to 2.0, but the verified version is 1.26.3. Please reference installation instruction in Section 2.3 for more details.
   1. Documentation: <https://numpy.org/doc/1.26/>
4. rskfd: This is a tool for instrument and data handling in R&S formats. Please reference installation instruction in Section 2.3 for more details.
   1. Documentation: <https://pypi.org/project/rskfd/>

On the SMW200A:

1. Realtime control interface (R&S®SMW-K503)
2. Realtime control interface with enhanced PDW rate and control PDWs (R&S®SMW-K504)

## What’s Included

Inside the folder attached, there are a few important files:

1. **PDWlist.csv** and **PDWlist\_verif.csv**– example csv files with PDWs and TCDWs.
   1. **PDWlist.csv** contains arb PDWs.
   2. **PDWlist\_verif.csv** does not contain arb PDWs.
2. **xdw\_file\_playback\_v2.py** – this is the script a user would run to generate the files needed for the import into the SMW200A. Running this script outputs the following files:
   1. **\*.ps\_def** file contains the PDWs and TCDWs. This file is always generated.
   2. **\*.ps\_adr** file contains the addresses of the waveform segments (addressed by PDWs). This file is generated when arb PDWs are included.
   3. **Pulse1.wv** and **Pulse2.wv** are example waveform files used in the csv file for use with the ‘xdw\_file\_playback\_v2.py’ script to generate the container waveform ‘**IQ\_expert.wv**’. This file is generated when arb PDWs are included.
3. **rsxdwstreaming-2.0.0-py2.py3-none-any.whl** - is the package file (wheel) for PDW Expert
4. **Doc** directory – Contains HTML formatted documentation for installation of the Rsxdwstreaming package

## Installation Instructions

### Pip

The Package Installer for Python (pip) is used to install the necessary packages for this script. If this is not installed on the PC system, please follow the instructions here: <https://www.geeksforgeeks.org/how-to-install-pip-on-windows/>

### Numpy

Numpy is a package used by the script for numeric operations. Install Numpy using the “pip” command. **Version 1.26.3** is verified to work with this version of the python script. **Versions 2.0 and later will not work.**

Run the following command in the command prompt.

>> python -m pip install numpy==1.26.3

### Rsxdwstreaming

The Rsxdwstreaming-2.0.0-py2.py3-none-any.whl package file (wheel) is for PDW Expert pulse building. This can be installed using the PIP command.

1. Save the rsxdwstreaming package on your computer, for example in c:\temp\
2. Start the command line WinKey + R, type cmd and hit ENTER
3. Change the working directory to the Python installation of your choice (adjust the user name and python version in the path):
   * cd c:\Users\John\AppData\Local\Programs\Python\Python37\Scripts
4. Install with the command: python -m pip install c:\temp\rsxdwstreaming-2.0.0-py2.py3-none-any.whl

Please see the doc/getting\_started.html for more information on this package.

### Rsfkd

The package rskfd is for instrument and data handling. Data handling mainly focuses on reading writing I/Q files in publich R&S formats. To install this, please use pip

>> python -m pip install rskfd

## Overview of Operation

The typical flow of operation would be as follows:

A white circle with black text

Description automatically generated

Figure 1: Typical Flow

Further details on each of these processes are described in Section 4.

## References

SMW200A User Manual: <https://scdn.rohde-schwarz.com/ur/pws/dl_downloads/dl_common_library/dl_manuals/gb_1/s/smw200a_1/SMW200A_UserManual_en_22.pdf>

SMW200A K503 Extended Sequencer User Manual: <https://www.rohde-schwarz.com/us/manual/r-s-smw200a-extended-sequencer-user-manual-manuals_78701-249480.html>

K503/-K504 PDW and TCDW ICD: <https://scdn.rohde-schwarz.com/ur/pws/dl_downloads/dl_application/application_notes/1gp133/ICD_PDW_TCDW_rev2.3.pdf>

# CSV Structure

The input to the script is a .csv with all the information about the PDWs and TCDWs.

The script searches for the following terms: Type, TOA, Mod, Arb, RF, Pulse Width, RF Freq, Level, Frequency Offset, Level Offset, Phase Offset, Freq Step, Chip Width, Bkr Code, Stuff, Burst PRI, Number of Pulses, Marker 1, Marker 2, and Marker 3. Columns that do not contain these terms will be ignored in the input file. **Columns that contain these terms, but are not inputs to the script, might confuse the parser.** For example, if you have a column titled “TOA (clicks)” that you use to calculate some intermediate values, the script might interpret that value as the TOA field.

Use as many columns as necessary for calculating values provided the naming convention is followed.

*Note: An .xlsx file will typically be used to build the list of PDWs and TCDWs since the fields are typically calculated from other fields using formulas. Please save your file as a .csv for an input list. This is so the raw values (not formulas) are saved in each cell.*

**Column Descriptions**

|  |  |  |  |
| --- | --- | --- | --- |
| Column Label in .csv | Supported Values | Unit | Description |
| Type | tdcw, pdw | N/A | Control RF Parameters and generate pulsed signals. |
| TOA | See ICD | seconds | Timestamp relative to scenario start trigger event. To avoid dropping PDWs, please allow for adequate time for the previous PDW to complete. |
| Mod | Rectangular,  Linear Chirp,  Tchirp,  Barker | N/A | In PDW, this is the type of pulse modulation. |
| RF | rffreq, rflevel, EOF | N/A | In TCDW, this specifies the type of TCDW. |
| Pulse Width |  | N/A | In PDW, this specifies the pulse width. |
| RF Freq | Instrument Specific | Hz | In TCDW, this specifies the center frequency of the modulation bandwidth. If a pulse is desired outside this bandwidth, a new TCDW with a new center frequency should be placed before the pulse is created. Please allow for 5ms settling time when modifying the frequency. |
| Level | Instrument Specific | dBm | In TDCW, this specifies the level of the instrument. When changing the level, please allow for 25ms settling time if the step attenuator is changed. |
| Frequency Offset | Instrument Specific | Hz | In PDW, the frequency offset is added to instrument RF frequency specified in the latest rffreq TCDW. |
| Level Offset | Instrument Specific | dBm | In PDW, the level offset is subtracted from instrument RF level specified in the latest rflevel TCDW. |
| Phase Offset | 0 - 360 | Degrees | In PDW, this is the phase offset |
| Freq Step | Instrument Specific | Hz/Sample | In Linear Chirp PDW, this specifies the frequency step in Hz/Sample. This value is dependent on the instrument clock. |
| Chip Width | See ICD | N/A | In Bkr Code PDW, this specifies the chip width of one Barker code chip |
| Bkr Code | 0-8 | N/A | In Bkr Code PDW, this is the code to select the type of Barker code. |
| Stuff | See ICD | N/A | Stuffing bits |
| Burst PRI | See ICD | seconds | In PDW, this is the Pulse Repetition Interval for burst |
| Number of Pulses | See ICD | N/A | In PDW, this is the total number of pulses for the PDW |
| Marker 1 | 1 or 0 | N/A | In PDW, this activates Marker 1 on the specified pulse. |
| Marker 2 | 1 or 0 | N/A | In PDW, this activates Marker 2 on the specified pulse. |
| Marker 3 | 1 or 0 | N/A | In PDW, this activates Marker 3 on the specified pulse. |

Please reference the included example .csv files PDWlist.csv and PDWlist\_verif.csv

*Note: Please ensure the last descriptor word in the list is an EOF (End of File) TCDW. This is vital for synchronization.*

# Workflow

A white circle with black text

Description automatically generated

## Build a list of PDWs and TCDWs in .xlsx Format

The first step in the process is to build a list of PDWs and TCDWs. This is typically done in .xlsx format so formulas can be leveraged for calculating various values such as TOAs. The .csv format can be used, but the Excel formulas will not be supported.

Each line in the .xlsx represents a pulse with a number of repetitions based on the “Number of Pulses” field.

All timing for the PDWs is based on the TOA (Time of Arrival) field. To avoid dropping PDWs, please allow for adequate time for the previous PDW to complete.

The last descriptor word in the file should be a EOF (End of File) TDCW. This ensures synchronization and terminates the list.

*Note: If you do not have a EOF TCDW as your last descriptor word, one will be appended by the script. However, it is best practice to have this included in the file.*

Please reference Section 3 for further details about the column descriptions and the included .csv files for examples.

## Save to .csv

The script needs comma separated text as an input. It does not support the more complex file format of .xlsx. Please treat the .xlsx as your master file and save a copy as .csv to feed into the script.

## Run Python Script

Run Command Prompt in the python\_package\_xDW\_SMW200A directory by right clicking and selecting “Open in Terminal”

A screenshot of a computer

Description automatically generated

Run the Python script:

>> python .\xdw\_file\_playback\_v2.py

A screenshot of a computer

Description automatically generated

Enter the .csv file name when prompted and hit “Enter”

A screenshot of a computer

Description automatically generated

Enter your comments which will be displayed upon selection of this file in the SMW200A. Press ENTER.

A screen shot of a computer

Description automatically generated

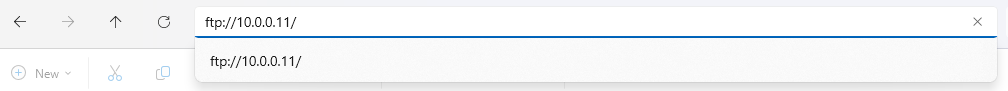
When using arb waveform PDWs, three files are generated.

1. **\*.ps\_def** file contains the PDWs and TCDWs.
2. **\*.ps\_adr** file contains the addresses of the waveform segments (addressed by PDWs). This file is generated when arb PDWs are included.
3. ‘**IQ\_expert.wv**’. This file is generated when arb PDWs are included.

When not using arb waveform PDWs, only **\*.ps\_def** file is generated.

## Transfer Script Output Files to the SMW200A

There are multiple ways to transfer files to and from the instrument. The FTP method was used here.

Type “ftp://<ip address>” in the Windows Explorer window. 

Type the username and password. Both are “instrument”

A screenshot of a computer screen

Description automatically generated

Copy and paste the three generated files under the “user” folder.

A screenshot of a computer

Description automatically generated

More methods and information can be found in the SMW200A User Manual <https://scdn.rohde-schwarz.com/ur/pws/dl_downloads/dl_common_library/dl_manuals/gb_1/s/smw200a_1/SMW200A_UserManual_en_22.pdf>

## Play the Files Using the Extended Sequencer

To play the file back, it is recommended to start from a PRESET.

1. Navigate to the “Baseband” -> “Extended Sequencer”A screenshot of a computer

   Description automatically generated
2. Under “Mode”, select “Playback from File”A screenshot of a computer

   Description automatically generated
3. Next, press “Definition File” and navigate to the “IQ\_expert.ps\_def” file.

A screenshot of a computer

Description automatically generated

1. Go to “Trigger In” tab and select the desired trigger method

A screenshot of a computer

Description automatically generated

1. Go to the “Marker” tab and ensure every marker type is selected to PDW. This ensures the SMW200A scans the incoming PDWs for Markers. The default selection is “Restart”.

A screenshot of a computer

Description automatically generated

1. Turn “RF On”

A screenshot of a computer

Description automatically generated

1. Under the “General” tab, turn ON Extended Sequencer

A screenshot of a computer

Description automatically generated

1. Under “Trigger In” tab, Select “Execute Trigger” (if in Armed Retrigger Mode),

A screenshot of a computer

Description automatically generated

# Optional Capabilities

## Continuous Playback

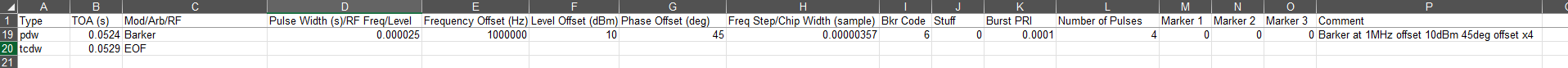
In the case that the user wants continuous looping playback of the file without the need to manually or externally trigger the SMW200A, there are a few modifications to the PDW list and SMW200A setup. This setup enables the SMW200A to retrigger playback itself by utilizing the marker functionality. Essentially, the setup requires a marker output at the end of the PDW list which will be looped back into the SMW200A for retriggering the file playback.

### PDW List Modifications

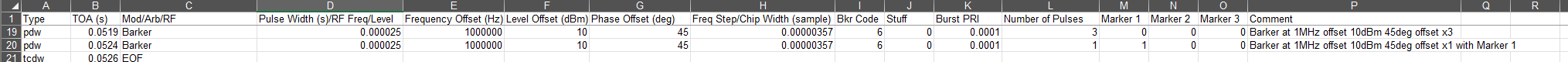
The last PDW in the list should have Marker 1 enabled. If the last PDW entry in the list has multiple “Number of Pulses” split it such that the last pulse has one “Number of Pulses”.

Example:

Before: The last PDW is a Barker with “Number of Pulses” = 4



After: The last PDW is split into two PDWs. The first has 3 pulses and no marker, and the second has 1 pulse with Marker 1 enabled.



### SMW200A Setup Modifications

* + - 1. Under the “Extended Sequencer” window in the baseband, navigate to the “Marker” tab and select “Global Connectors”

A screenshot of a computer

Description automatically generated

* + - 1. Under the “Routing” tab, ensure User 1 is routed to “Baseband A Marker 1” and User 3 is routed to “Global Trigger 1”.

A screenshot of a computer

Description automatically generated

* + - 1. Using a BNC connector, connect User 1 to User 3 on the front panel of the SMW200A

A close-up of a device

Description automatically generated

* + - 1. Under the “Trigger In” tab in the “Extended Sequencer”, select Mode -> Retrigger and Source -> External Global Trigger 1. This allows the User 3 port to trigger the playback of the file.

A screenshot of a computer

Description automatically generated

Now, the file will trigger itself to replay at the beginning on the last PDW of the list.

# Q&A

TODO