# TWM correction datasets reference manual

V0.4, 2018-12-04

Following text describes how to create the correction datasets for digitizer and transducers for TWM tool [1].

Table of contents[Table of contents 1](#_Toc531694777)

[1.1 Resources 1](#_Toc531694778)

[1.2 Abbreviations 2](#_Toc531694779)

[1.3 Introduction 2](#_Toc531694780)

[1.3.1 CSV tables 2](#_Toc531694781)

[1.3.1.1 1D CSV table format 2](#_Toc531694782)

[1.3.1.2 2D CSV table format 3](#_Toc531694783)

[1.3.2 Correction model 3](#_Toc531694784)

[1.4 Transducer corrections 5](#_Toc531694785)

[1.4.1 Transducer correction items 6](#_Toc531694786)

[1.4.1.1 Nominal ratio 6](#_Toc531694787)

[1.4.1.2 Amplitude and phase transfer (optional) 6](#_Toc531694788)

[1.4.1.3 Transducer SFDR value (optional) 7](#_Toc531694790)

[1.4.1.4 Transducer low-side RVD impedance (optional) 8](#_Toc531694791)

[1.4.1.5 Transducer high-side output terminal series impedance (optional) 8](#_Toc531694792)

[1.4.1.6 Transducer low-side output terminal series impedance (optional) 9](#_Toc531694793)

[1.4.1.7 Transducer output terminals mutual inductance (optional) 9](#_Toc531694794)

[1.4.1.8 Transducer output terminals shunting admittance (optional) 10](#_Toc531694795)

[1.4.1.9 Optional buffer output series impedance (optional) 10](#_Toc531694796)

[1.4.1.10 Cable(s) series impedance (optional) 11](#_Toc531694797)

[1.4.1.11 Cable(s) shunting admittance (optional) 11](#_Toc531694798)

[1.5 Digitizer corrections 11](#_Toc531694799)

[1.5.1 Digitizer correction table format 12](#_Toc531694800)

[1.5.2 Digitizer corrections 14](#_Toc531694801)

[1.5.2.1 Inter-channel time-shift correction (optional) 14](#_Toc531694802)

[1.5.2.2 Timebase correction (optional) 15](#_Toc531694803)

[1.5.2.3 Inter-channel crosstalk 15](#_Toc531694804)

[1.5.3 Channel corrections 15](#_Toc531694805)

[1.5.3.1 Nominal gain (optional) 15](#_Toc531694806)

[1.5.3.2 Gain frequency transfer (optional) 15](#_Toc531694807)

[1.5.3.3 Phase frequency transfer (optional) 16](#_Toc531694808)

[1.5.3.4 DC offset (optional) 17](#_Toc531694809)

[1.5.3.5 Aperture correction (optional) 17](#_Toc531694810)

[1.5.3.6 SFDR value (optional) 17](#_Toc531694811)

[1.5.3.7 RMS jitter (optional) 18](#_Toc531694812)

[1.5.3.8 Input admittance (optional) 18](#_Toc531694813)

## Resources

1. TWM tool, url: <https://github.com/smaslan/TWM>
2. INFO-STRINGS, url: <https://github.com/KaeroDot/info-strings>
3. QWTB toolbox, url: <https://qwtb.github.io/qwtb/>
4. A232 Algorithms exchange format, url: [https://github.com/smaslan/TWM/tree/master/doc/A232 Algorithm Exchange Format.docx](https://github.com/smaslan/TWM/tree/master/doc/A232%20Algorithm%20Exchange%20Format.docx)

## Abbreviations

TWM – Traceable power quality WattMeter

SFDR – Spurious Free Dynamic Range

## Introduction

All correction files are based on the combination of INFO-STRINGS library [2] and ordinary CSV files. The corrections are loaded automatically by the TWM tool and passed to the PQ algorithm wrapped in the QWTB toolbox [3]. The following text shows the formats of the correction datasets, behavior of the TWM correction loader and naming of the correction values and tables that will be passed to the QWTB algorithm.

### CSV tables

Many of the correction tables in the TWM tool are stored as a CSV tables separated by semicolon “;”. This is e.g. the case of dependencies, such frequency transfers. This solution was chosen to ensure flexibility and easy editing for the users. All the tables must have unified format.

#### 1D CSV table format

1D dependence CSV table with three quantities **A**, **B** and **C** dependent on axis **Y**:

|  |  |  |  |
| --- | --- | --- | --- |
| Comment |  |  |  |
| Y | A | B | C |
| y1 | a1 | b1 | c1 |
| y2 | a2 | b2 | c2 |
| y3 | a3 | b3 | c3 |

The “Comment” is any text string that describes content of the table. The row is mandatory even if the comment is not required!

Any table can contain empty values:

|  |  |  |  |
| --- | --- | --- | --- |
| Comment |  |  |  |
| Y | A | B | C |
| y1 | a1 |  | c1 |
| y2 |  | b2 | c2 |
| y3 | a3 | b3 |  |

The missing value **a2** will be interpolated from **a1** and **a3** by the loader. However, missing value **b1** and **c3** will be loaded as **NaN** because they are at the boundary of the table and extrapolation is disabled since the uncertainty of extrapolation cannot be properly evaluated.

The 1D table can be also independent on the axis **Y** if the table has just one data row and empty **Y** axis, i.e. missing value **y1**:

|  |  |  |  |
| --- | --- | --- | --- |
| Comment |  |  |  |
| Y | A | B | C |
|  | a1 | b1 | c1 |

All TWM functions will ignore the axis **Y** and will assume the values **a1**, **b1**, **c3** for any value of **Y**.

#### 2D CSV table format

TWM also supports 2D tables dependent on two axes **X** and **Y**:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Comment |  |  |  |  |  |  |
|  | A | A | B | B | C | C |
| Y \ X | x1 | x2 | x1 | x2 | x1 | x2 |
| y1 | a11 | a12 | b11 | b12 | c11 | c12 |
| y2 | a21 | a22 | b21 | b22 | c21 | c22 |
| y3 | a31 | a32 | b31 | b32 | c31 | c32 |

The table can contain any number of quantities (**A**, **B**, **C**, …). **Y** axis is identical as in 1D tables. **X** axis is horizontal and its values **x1**, **x2**, … are repeated for each quantity. All quantities must have identical number of **X** values. The 2D table can be independent on **Y** axis:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Comment |  |  |  |  |  |  |
|  | A | A | B | B | C | C |
| Y \ X | x1 | x2 | x1 | x2 | x1 | x2 |
|  | a11 | a12 | b11 | b12 | c11 | c12 |

The 2D table can be also independent on **X** axis if all **X** values are empty:

|  |  |  |  |
| --- | --- | --- | --- |
| Comment |  |  |  |
|  | A | B | C |
| Y \ X |  |  |  |
| y1 | a11 | b11 | c11 |
| y2 | a21 | b21 | c21 |
| y3 | a31 | b31 | c31 |

Eventually, the 2D table can be independent on both axes **X**, **Y**:

|  |  |  |  |
| --- | --- | --- | --- |
| Comment |  |  |  |
|  | A | B | C |
| Y \ X |  |  |  |
|  | a11 | b11 | c11 |

### Correction model

The TWM tool and the implemented algorithms perform corrections to the errors introduced by the digitizer and transducer. However, when the transducer is connected to the digitizer via cable, the transducer’s transfer will be affected by the loading effects due to finite input impedance of the digitizer and capacitance of the cable. This effect can be corrected if the lumped impedance model of the transducer terminals, cables and digitizer is known. Thus a special function dealing with this problem for single-ended and differential connection of the transducer to digitizer channels was developed and each algorithm should employ it. The function is able to calculate corrections in four different configurations shown in Figure 0‑1, Figure 0‑2, Figure 0‑3 and Figure 0‑4.

TWM will choose single-ended or differential based on the configuration of the transducer corrections. The buffered mode is enabled by including the buffer output impedance “**Zbuf**” to the transducer corrections (see details below).

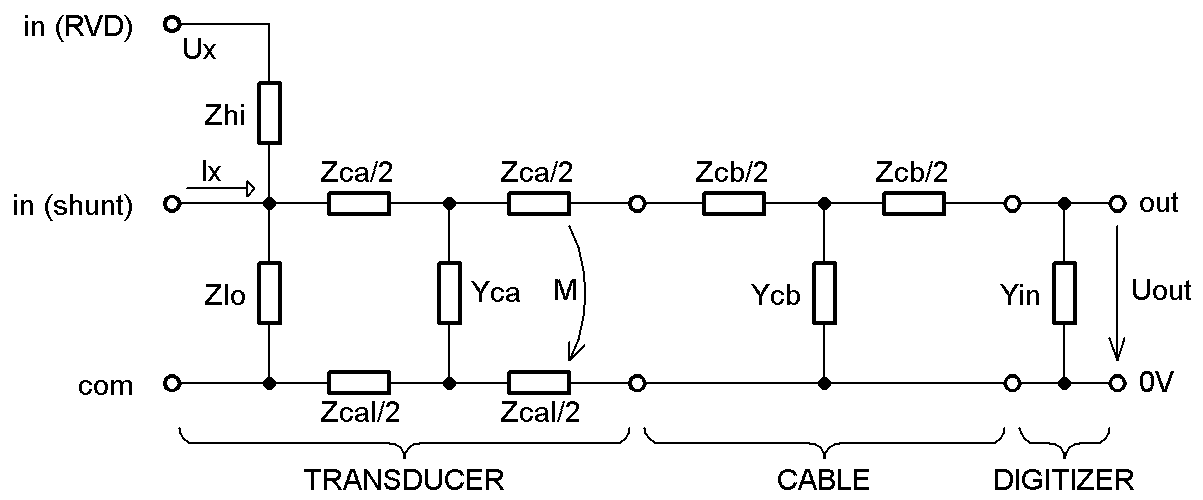


Figure 0‑1: Single-ended direct connection (no buffer).

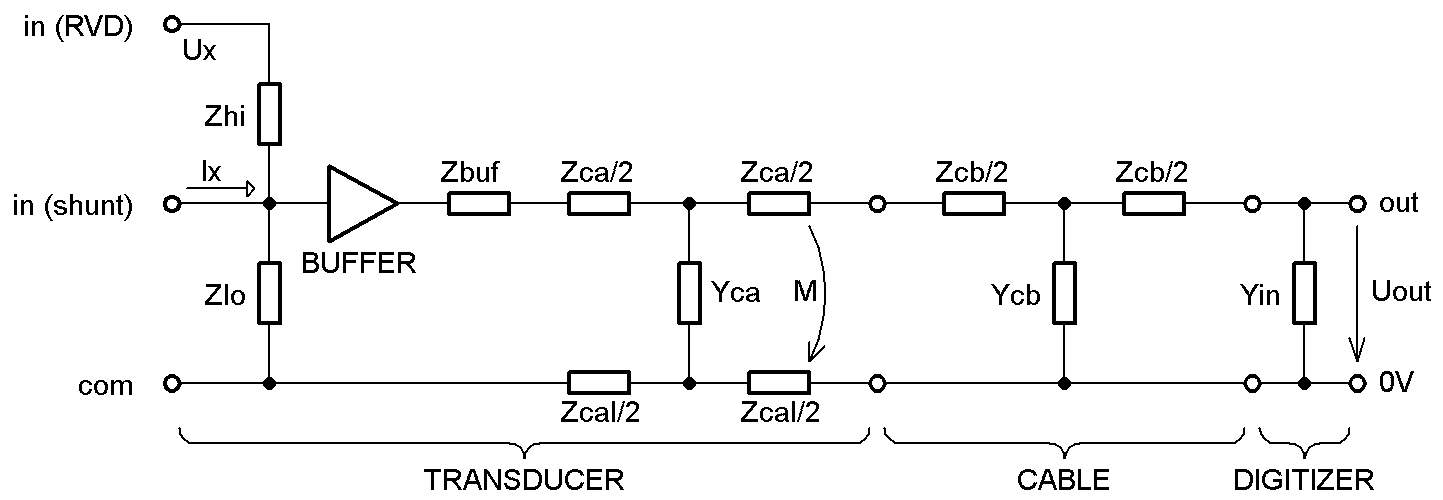


Figure 0‑2: Single-ended buffered connection.

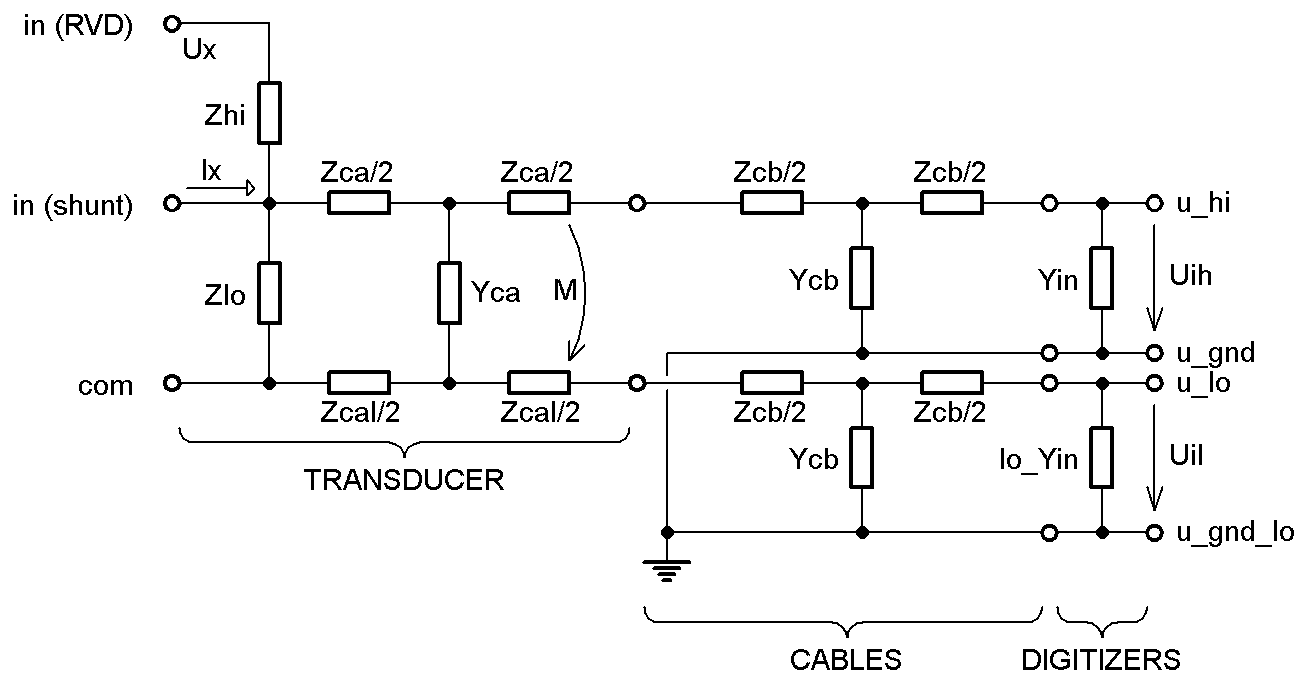


Figure 0‑3: Direct differential connection (no buffer).

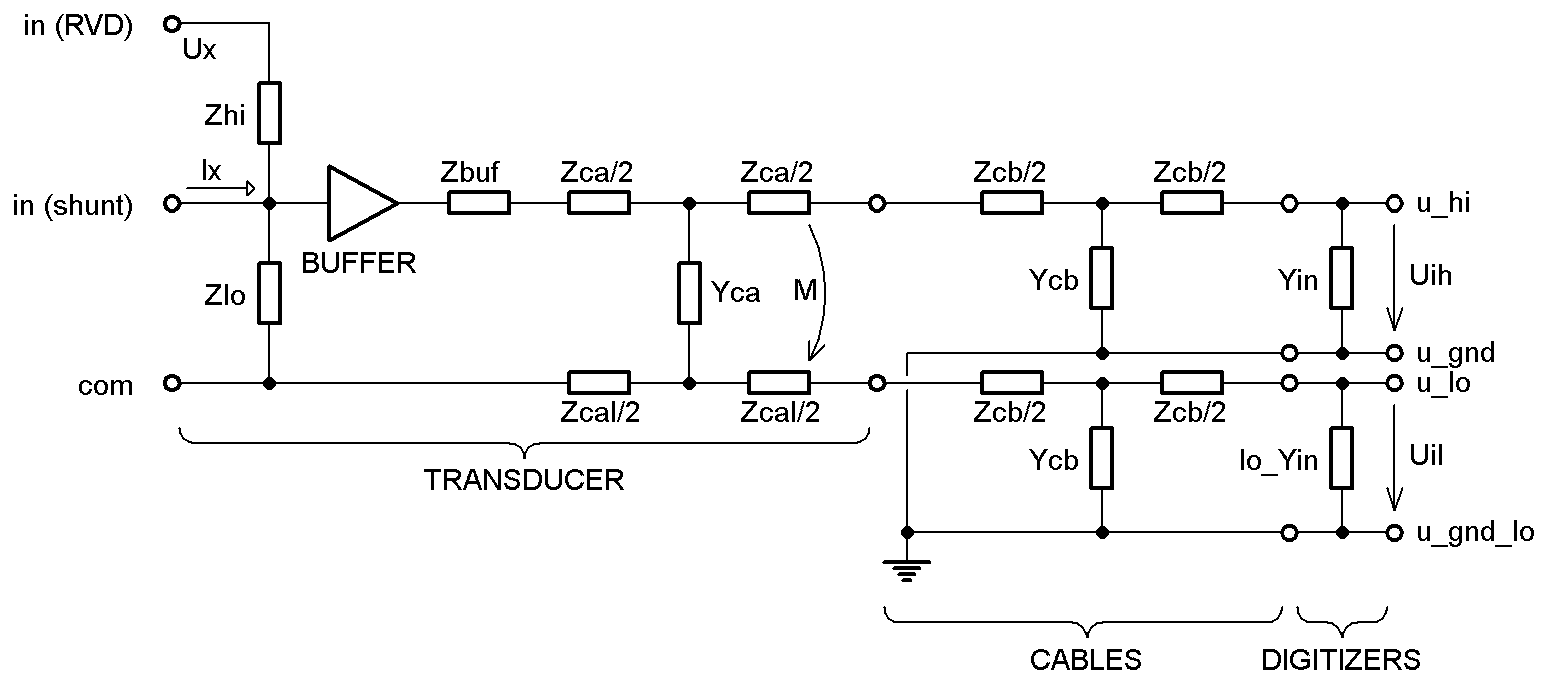


Figure 0‑4: Buffered differential connection.

The impedance components “**Yin**” (and “**lo\_Yin**”) comes from the digitizer channel corrections, whereas the rest of the model components are loaded from the transducer correction file.

## Transducer corrections

TWM recognizes two types of transducer corrections: “**divider**” and “**shunt**”. Format of the correction file is identical for both. File starts with the identifier “**type**” which defines the transducer type. “**name**” and “**serial number**” is description of the transducer. Optional item “**linked to digitizer channel**” can restrict the use of the transducer correction file to particular digitizer channel which may be needed when the transducer and digitizer channel were calibrated together. Next, the main correction data follows (description below). Example of the file for a shunt:

*// type of the correction:*

type:: shunt

*// name of the transducer:*

name:: Current shunt 1A

*// serial number of the transducer:*

serial number:: CMI/1A/1/13

*// identifier of the channel of the digitizer if the transducer was calibrated together with the digitizer:*

*// note: leave empty or remove if not needed*

linked to digitizer channel:: HP3458A, sn. MY45053095

*// nominal/DC ratio: V/A for shunt, Vin/Vout for divider:*

nominal ratio:: 0.600005

nominal ratio uncertainty:: 0.000009

*// frequency transfer of the transducer - amplitude (input/output):*

*// 2D CSV table:*

*// y-axis: frequency*

*// x-axis: input rms value*

*// quantity 1: in/out transfer values*

*// quantity 2: absolute uncertainties*

amplitude transfer path:: csv\tfer\_amp.csv

*// frequency transfer of the transducer - phase (input - output):*

*// 2D CSV table identical format to amp. transfer.*

phase transfer path:: csv\tfer\_phi.csv

*// frequency dependence of impedance of the low-side resistor of RVD:*

*// 2D CSV table,*

*// y-axis: fundamental frequency*

*// x-axis: fundamental amplitude*

*// quantities order: sfdr [dB], u(sfdr)*

sfdr path:: csv\sfdr.csv

*// --- loading correction components ---*

*// frequency dependence of series impedance of transducer’s high-side terminal:*

*// 1D CSV table, y-axis: frequency, quantities order: Rs, Ls, u(Rs), u(Ls)*

output terminals series impedance path:: csv\Zca.csv

*// frequency dependence of series impedance of transducer’s low-side terminal:*

*// 1D CSV table, y-axis: frequency, quantities order: Rs, Ls, u(Rs), u(Ls)*

output terminals series impedance path (low-side):: csv\Zcal.csv

*// frequency dependence of mutual inductance between transducer’s terminals:*

*// 1D CSV table, y-axis: frequency, quantities order: M, u(M)*

output terminals mutual inductance path:: csv\Zcam.csv

*// frequency dependence of loss admittance between the transducer’s terminals:*

*// 1D CSV table, y-axis: frequency, quantities order: Cp, D, u(Cp), u(D)*

output terminals shunting admittance path:: csv\Yca.csv

*// frequency dependence of series impedance of the cable to digitizer input:*

*// 1D CSV table, y-axis: frequency, quantities order: Rs, Ls, u(Rs), u(Ls)*

output cable series impedance path:: csv\Zcb.csv

*// frequency dependence of shunting admittance of the cable to digitizer input:*

*// 1D CSV table, y-axis: frequency, quantities order: Cp, D, u(Cp), u(D)*

output cable shunting admittance path:: csv\Ycb.csv

*// frequency dependence of impedance of the low-side resistor of RVD:*

*// 1D CSV table, y-axis: frequency, quantities order: Rp, Cp, u(Rp), u(Cp)*

rvd low side impedance path:: csv\Z\_low.csv

*// frequency dependence of impedance of the buffer output series impedance (leave out to disable buffer!):*

*// 1D CSV table, y-axis: frequency, quantities order: Rs, Ls, u(Rs), u(Ls)*

buffer output series impedance path:: csv\Z\_buf.csv

### Transducer correction items

Following paragraphs describe particular correction components. It will always show formats of the correction data and naming of the correction data quantities that will be passed to the QWTB.

#### Nominal ratio

Nominal ratio item “**nominal ratio**” is scalar real value that defines nominal (typically DC) ratio of the transducer. For shunt it is value in Ohms. For divider it is input/output ratio. The value has also absolute uncertainty “**nominal ratio uncertainty**”. Both values are mandatory.

Note it is possible to use these items to store nominal gain for any frequency, e.g. 50 Hz. It is not restricted to DC. The relative Amplitude transfer will be always relative to this value. So the choice is up to the user.

#### Amplitude and phase transfer (optional)

“**amplitude transfer path**” and “**phase transfer path**” are paths to the CSV files with 2D frequency-amplitude transfers relative to the nominal ratio. Both amplitude and phase transfers can have different frequency and amplitude dependency axes. If these corrections(s) are not defined, TWM will use value of 1 for “**amplitude transfer path**” and value 0 for “**phase transfer path**”. User should always define the correction down to zero frequency in order to make algorithms requiring DC value work! It may be also needed to define frequency dependence up to Nyquist frequency for the FFT based algorithms.

Format of the CSV table for amplitude transfer:

|  |  |
| --- | --- |
| x-axis: | input rms value [V] or [A] |
| y-axis: | frequency [Hz] |
| Quantities: | gain – relative gain  u(gain) – absolute std. uncertainty of gain |

Format of the CSV table for phase transfer:

|  |  |
| --- | --- |
| x-axis: | input rms value [V] or [A] |
| y-axis: | frequency [Hz] |
| Quantities: | phi – absolute phase shift  u(phi) – absolute std. uncertainty of phi |

Note the x-axis is dependent on input voltage (or current), not the output one! The correction loader will always combine “**nominal ratio**” and “**amplitude transfer path**” into a single absolute correction table:

abs gain = nominal ratio \* gain

abs gain uncertainty = sqrt((nominal ratio uncertainty)^2 + u(gain)^2)

Note the “**abs gain**” and its uncertainty will be automatically inverted by TWM for a shunt so the “**abs gain**” is always a ratio of measured input quantity (voltage or current) and transducer output voltage. Which means the “**gain**” for divider is relative dependence of input-to-output division ratio, whereas “**gain**” for shunt is relative dependence of impedance of the shunt!

The “**abs gain**” is always passed to the QWTB as quantities:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*tr\_gain\_f.v | [] | Frequency axis [Hz] |
| \*tr\_gain\_a.v | [] | Input RMS value axis [V] or [A] |
| \*tr\_gain.v  \*tr\_gain.u | 1  0 | Gain  Abs. std. uncertainty of gain |

\* - transducer prefix [4] (i.e. “u\_” or “i\_” or nothing)

The transducer’s phase shift is always passed to the QWTB as quantities:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*tr\_phi\_f.v | [] | Frequency axis [Hz] |
| \*tr\_phi\_a.v | [] | Input RMS value axis [V] or [A] |
| \*tr\_phi.v  \*tr\_phi.u | 0  0 | Phase correction [rad]  Abs. std. uncertainty of phase correction [rad] |

\* - transducer prefix [4] (i.e. “u\_” or “i\_” or nothing)

Follows example of shunt correction data. Lets assume a shunt has following impedance characteristic:

|  |  |  |
| --- | --- | --- |
| *f* | *Z [Ω]* | *u(Z) [Ω]* |
| DC | 0.600 000 | 0.000 001 |
| 1 kHz | 0.600 006 | 0.000 003 |
| 10 kHz | 0.600 060 | 0.000 030 |
| 100 kHz | 0.600 600 | 0.000 060 |

Then the shunt can be described by nominal ratio (DC value of resistance):  
**nominal ratio = 0.600000  
nominal ratio uncertainty = 0.000001**and by the relative amplitude transfer “**amplitude transfer path**” CSV table:

|  |  |  |
| --- | --- | --- |
| *My shunt description* | | |
|  | *gain* | *u(gain)* |
| *f \ rms* |  |  |
| DC | 1.000000 | 0.000000 |
| 1 kHz | 1.000010 | 0.000005 |
| 10 kHz | 1.000100 | 0.000050 |
| 100 kHz | 1.001000 | 0.000100 |

#### Transducer SFDR value (optional)

Defines effects of distortion of the transducer. The “**sfdr path**” is path to the 2D CSV file with measured SFDR values dependent on amplitude and frequency if fundamental frequency of the signal. The values are in [dB]. Note the values are positive, i.e.: 120 dB means max spur amplitude is *A0\*10^-(120/20)*.

Format of the CSV table:

|  |  |
| --- | --- |
| x-axis: | input amplitude of fundamental frequency value [V] or [A] |
| y-axis: | fundamental frequency [Hz] |
| Quantities: | sfdr – SFDR value [dB] |

The QWTB quantity naming:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*tr\_sfdr\_f.v | [] | Frequency axis [Hz] |
| \*tr\_sfdr\_a.v | [] | Input amplitude axis [V] or [A] |
| \*tr\_sfdr.v | 180 | SFDR value [dB] |

\* - transducer prefix [4] (i.e. “u\_” or “i\_” or nothing)

#### Transducer low-side RVD impedance (optional)

1D CSV table “**rvd low side impedance path**” defines rough impedance of the low-side resistor for RVDs. This value is needed only for RVD and it is used to calculate loading effect of the cable and digitizer input to the transfer. It is a “**Zlo**” component in the connection diagram. Typically it is not necessary to calibrate the value to uncertainty below 0.1 % if the resistance of the RVD is up to few hundred ohms and total impedance is above 1 MΩ. For a shunt the value is ignored as the impedance of shunt can be calculated from the absolute complex transfer.

Format of the CSV table:

|  |  |
| --- | --- |
| y-axis: | frequency [Hz] |
| Quantities: | Rp – parallel resistance [Ω]  Cp – parallel capacitance [F]  u(Rp) – absolute std. uncertainty Rp  u(Cp) – absolute std. uncertainty Cp |

The QWTB quantity naming:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*tr\_Zlo\_f.v | [] | Frequency axis [Hz] |
| \*tr\_Zlo\_Rp.v  \*tr\_Zlo\_Rp.u  \*tr\_Zlo\_Cp.v  \*tr\_Zlo\_Cp.u | 1e3  0  0  0 | Rp value [Ω]  Abs. std. uncertainty of Rp [Ω]  Cp value [F]  Abs. std. uncertainty of Cp [F] |

\* - transducer prefix [4] (i.e. “u\_” or “i\_” or nothing)

#### Transducer high-side output terminal series impedance (optional)

1D CSV table “**output terminals series impedance path**” is estimate of the series impedance of the transducer’s high-side output terminal (component “**Zca**” in the correction diagram). It is part of the transducer loading corrections. The value is usually not measurable, but at least its uncertainty should be estimated in order take the loading effect into the uncertainty budget.

Format of the CSV table:

|  |  |
| --- | --- |
| y-axis: | frequency [Hz] |
| Quantities: | Rs – series resistance [Ω]  Ls – series inductance [H]  u(Rs) – absolute std. uncertainty Rs  u(Ls) – absolute std. uncertainty Ls |

The QWTB quantity naming:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*tr\_Zca\_f.v | [] | Frequency axis [Hz] |
| \*tr\_Zca\_Rs.v  \*tr\_Zca\_Rs.u  \*tr\_Zca\_Ls.v  \*tr\_Zca\_Ls.u | 1e-9  0  1e-12  0 | Rs value [Ω]  Abs. std. uncertainty of Rs [Ω]  Ls value [H]  Abs. std. uncertainty of Ls [H] |

\* - transducer prefix [4] (i.e. “u\_” or “i\_” or nothing)

#### Transducer low-side output terminal series impedance (optional)

1D CSV table “**output terminals series impedance path (low-side)**” is estimate of the series impedance of the transducer’s low-side output terminal (component “**Zcal**” in the correction diagram). It is part of the transducer loading corrections. The value is usually not measurable, but at least its uncertainty should be estimated in order take the loading effect into the uncertainty budget. Note for the single-ended connection this component can be part of the high-side impedance “**Zca**” and this correction can be left unassigned.

Format of the CSV table:

|  |  |
| --- | --- |
| y-axis: | frequency [Hz] |
| Quantities: | Rs – series resistance [Ω]  Ls – series inductance [H]  u(Rs) – absolute std. uncertainty Rs  u(Ls) – absolute std. uncertainty Ls |

The QWTB quantity naming:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*tr\_Zcal\_f.v | [] | Frequency axis [Hz] |
| \*tr\_Zcal\_Rs.v  \*tr\_Zcal\_Rs.u  \*tr\_Zcal\_Ls.v  \*tr\_Zcal\_Ls.u | 1e-9  0  1e-12  0 | Rs value [Ω]  Abs. std. uncertainty of Rs [Ω]  Ls value [H]  Abs. std. uncertainty of Ls [H] |

\* - transducer prefix [4] (i.e. “u\_” or “i\_” or nothing)

#### Transducer output terminals mutual inductance (optional)

1D CSV table “**output terminals mutual inductance**” is estimate of the mutual inductance between the transducer’s output terminals (component “**M**” in the correction diagram). It is part of the transducer loading corrections. The value is usually not measurable, but at least its uncertainty should be estimated in order take the loading effect into the uncertainty budget. Note for the single-ended connection the value of impedance “**M**”, “**Zcal**” and “**Zca**” can be combined to correction “**Zca**”:

Zca = Zca + Zcal – j\*4\*pi\*M

In that case this correction and low-terminal series impedance correction “**Zca**” can be left empty. However for differential mode the values of “**Zca**”, “**Zcal**” and “**M**” should be at least estimated especially for high frequency measurements.

Format of the CSV table:

|  |  |
| --- | --- |
| y-axis: | frequency [Hz] |
| Quantities: | M – mutual inductance [H]  u(M) – absolute std. uncertainty M |

The QWTB quantity naming:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*tr\_Zcam\_f.v | [] | Frequency axis [Hz] |
| \*tr\_Zcam.v  \*tr\_Zcam.u | 1e-12  0 | M value [H]  Abs. std. uncertainty of M [H] |

\* - transducer prefix [4] (i.e. “u\_” or “i\_” or nothing)

#### Transducer output terminals shunting admittance (optional)

1D CSV table “**output terminals shunting admittance path**” is estimate of the shunting admittance between the transducer’s output terminals (component “**Yca**” in the correction diagram). It is part of the transducer loading corrections. The value is usually not measurable, but at least its uncertainty should be estimated in order take the loading effect into the uncertainty budget.

Format of the CSV table:

|  |  |
| --- | --- |
| y-axis: | frequency [Hz] |
| Quantities: | Cp – parallel capacitance [F]  D – loss tangent [-]  u(Cp) – absolute std. uncertainty Cp  u(D) – absolute std. uncertainty D |

The QWTB quantity naming:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*tr\_Yca\_f.v | [] | Frequency axis [Hz] |
| \*tr\_Yca\_Cp.v  \*tr\_Yca\_Cp.u  \*tr\_Yca\_D.v  \*tr\_Yca\_D.u | 1e-15  0  0  0 | Cp value [F]  Abs. std. uncertainty of Cp [F]  D value [-]  Abs. std. uncertainty of D [-] |

\* - transducer prefix [4] (i.e. “u\_” or “i\_” or nothing)

#### Optional buffer output series impedance (optional)

1D CSV table “**buffer output series impedance path**” is effective series output impedance of the buffer placed between transducer and output terminals “**Zca**”/”**Zcal**”. The buffer presence is identified by this correction, so **do not assign it to tell TWM that no buffer is used**.

Format of the CSV table:

|  |  |
| --- | --- |
| y-axis: | frequency [Hz] |
| Quantities: | Rs – series resistance [Ω]  Ls – series inductance [H]  u(Rs) – absolute std. uncertainty Rs  u(Ls) – absolute std. uncertainty Ls |

The QWTB quantity naming:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*tr\_Zbuf\_f.v | [] | Frequency axis [Hz] |
| \*tr\_Zbuf\_Rs.v  \*tr\_Zbuf\_Rs.u  \*tr\_Zbuf\_Ls.v  \*tr\_Zbuf\_Ls.u | 0  0  0  0 | Rs value [Ω]  Abs. std. uncertainty of Rs [Ω]  Ls value [H]  Abs. std. uncertainty of Ls [H] |

\* - transducer prefix [4] (i.e. “u\_” or “i\_” or nothing)

#### Cable(s) series impedance (optional)

1D CSV table “**output cable series impedance path**” is effective series impedance of the cable between transducer and digitizer (component “**Zcb**” in the correction diagram). It is part of the transducer loading corrections. In differential mode both high- and low-side cables are expected to be identical! Note the cable correction can be omitted if the transducer was calibrated together with the cable.

Format of the CSV table:

|  |  |
| --- | --- |
| y-axis: | frequency [Hz] |
| Quantities: | Rs – series resistance [Ω]  Ls – series inductance [H]  u(Rs) – absolute std. uncertainty Rs  u(Ls) – absolute std. uncertainty Ls |

The QWTB quantity naming:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*Zcb\_f.v | [] | Frequency axis [Hz] |
| \*Zcb\_Rs.v  \*Zcb\_Rs.u  \*Zcb\_Ls.v  \*Zcb\_Ls.u | 1e-9  0  1e-12  0 | Rs value [Ω]  Abs. std. uncertainty of Rs [Ω]  Ls value [H]  Abs. std. uncertainty of Ls [H] |

\* - transducer prefix [4] (i.e. “u\_” or “i\_” or nothing)

#### Cable(s) shunting admittance (optional)

1D CSV table “**output cable shunting admittance path**” is estimate of the shunting admittance between the transducer’s output terminals (component “**Ycb**” in the correction diagram). It is part of the transducer loading corrections. In differential mode both high- and low-side cables are expected to be identical! Note the cable correction can be omitted if the transducer was calibrated together with the cable.

Format of the CSV table:

|  |  |
| --- | --- |
| y-axis: | frequency [Hz] |
| Quantities: | Cp – parallel capacitance [F]  D – loss tangent [-]  u(Cp) – absolute std. uncertainty Cp  u(D) – absolute std. uncertainty D |

The QWTB quantity naming:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*Ycb\_f.v | [] | Frequency axis [Hz] |
| \*Ycb\_Cp.v  \*Ycb\_Cp.u  \*Ycb\_D.v  \*Ycb\_D.u | 1e-15  0  0  0 | Cp value [F]  Abs. std. uncertainty of Cp [F]  D value [-]  Abs. std. uncertainty of D [-] |

\* - transducer prefix [4] (i.e. “u\_” or “i\_” or nothing)

## Digitizer corrections

Digitizer correction dataset consists of the two parts:

1. Definition of the whole digitizer (interchannel corrections),
2. Definition of the particular channels (corrections that are independent to another channel or HW).

### Digitizer correction table format

The format of every correction table for the digitizer and its channels is identical. The format was designed so the so it allows following:

1. Filtering the correction file by attributes of the digitizer
2. Automatic selection or interpolation of the correction data by the configuration (parameters) of the digitizer.
3. Loading either embedded numeric tables or CSV tables.

The correction data are always enclosed in the INFO file section, where the “my correction name” is the name of the correction:

**#startsection**:: my correction name

*// correction content*

**#endsection**:: my correction name

The correction must contain at least one item – the matrix with the correction data named “**value**”:

**#startsection**:: my correction name

*// up to 2D matrix with the list of values of the correction:*

**#startmatrix**:: value

**0.0; 0.10000; 0.20000**

**0.0; 0.10000; 0.20000**

**0.0; 0.01000; 0.02000**

**#endmatrix**:: value

**#endsection**:: my correction name

The value may be scalar, vector or 2D matrix of real numbers. If nothing else is present in the correction section, the correction loader will load the table of values as it is and will pass it to the QWTB algorithm under quantity name defined by the particular correction (see particular correction descriptions). The value may also contain relative path(s) to the CSV tables (single, vector of CSV files or 2D matrix of CSV files) that contains CSV table with 1D or 2D dependence (see introduction). If the “**value**” contains real numbers, the TWM can also load associated absolute std. uncertainty from complementary matrix “**uncertainty**” (for CSV file mode the uncertainty is part of the CSV table):

**#startsection**:: my correction name

*// ........*

**#startmatrix**:: uncertainty

**0.00; 0.00010; 0.00020**

**0.00; 0.00010; 0.00020**

**0.00; 0.00011; 0.00022**

**#endmatrix**:: uncertainty

**#endsection**:: my correction name

The correction loader can automatically select or interpolate between the values in the matrices “**value**” (and “**uncertainty**”) based on the value of any attribute (parameter) of the digitizer that is present in the measurement header. This is useful whenever the correction value depends on some setting of the digitizer. For example the measurement header always contains parameter “**voltage ranges [V]**” with range of the digitizer so it is possible to insert following section to the correction:

**#startsection**:: my correction name

*// ........*

*// --- List of parameters on which the correction values depends: ---*

*// primary parameter (remove if not used):*

**#startsection**:: primary parameter

*// name of the HW parameter:*

*// note: it must be exact name of the parameter that appears in measurement header*

name:: voltage ranges [V]

*// is this parameter interpolable?*

*// note: set to 0 or remove if not interpolable*

interpolable:: 0

*// list of supported values of a primary parameter on which the correction depends:*

*// eg.: range of the digitizer*

**#startmatrix**:: value

**1**

**10**

**100**

**#endmatrix**:: value

**#endsection**:: primary parameter

**#endsection**:: my correction name

The section “**primary parameter**” defines vertical axis of interpolation (selection) of the “**value**” matrix. I.e. for range value “10”, it will select second row of table “**value**”. The “**value**” of the interpolation parameter may be string as well as numeric. If it is numeric and the “**interpolable**” is non-zero, the loader will interpolate the “**value**” vertically per columns. If section “**secondary parameter**” is added to the correction section, it will do the same as “**primary parameter**” except in horizontal direction. Note each parameter reduces size of the “**value**” matrix by one dimension by the interpolation/selection, so when it is 2D matrix and one parameter is used, it will be interpolated to 1D vector (horizontal or vertical). If two parameters are defined, it will be interpolated to scalar value.

Note if the “**value**” matrix of the correction data is table of paths to CSV files, the loader will select/interpolate between the CSV tables as well. It will first interpolate content of all involved CSV tables to identical x- and y-axes, then it will interpolate between the tables by the “**primary parameter**” and “**secondary parameter**”, so the result is one CSV table. This is useful for example for the frequency dependence of the digitizer channel gain which may be dependent on the aperture and range of the digitizer.

Last supported feature of the correction section is filtering the corrections by attribute of the digitizer. Let’s assume the measurement header contains parameter “**sampling mode**”. The filter may look like this:

**#startsection**:: my correction name

*// ........*

*// --- Filtering of the correction by HW attributes: ---*

*// this is the list of channel specific attributes for which the correction is valid*

*// anything put here will be checked with the digitizer setup stored in the header file*

*// of the measurement and if it does not match, the loader will return an error*

**#startmatrix**:: valid for attributes

sampling mode

**#endmatrix**:: valid for attributes

*// list of allowed values of attribute 1 (eg.: sampling mode):*

**#startmatrix**:: sampling mode

DSDC

DSAC

**#endmatrix**:: sampling mode

**#endsection**:: my correction name

The “**valid for attributes**” list defines list of measurement header attributes which are used for filtering. Each attribute has its own list of allowed string values. In this case matrix “**sampling mode**” contains values “DSDC” and “DSAC”. If any other value is found in the measurement header or the “**sampling mode**” attribute is not found at all, the loader will return an error, which signalizes the correction is not compatible with selected HW and its configuration.

### Digitizer corrections

The digitizer correction defines the digitizer as a whole system. It contains list of all channels (e.g. sampling multimeters used in the setup). It also contains correction data which are somehow defines relation between multiple channels, such as interchannel timeshift. Example of the digitizer correction header INFO file is show in the following text:

*// correction type:*

type:: digitizer

*// description of the digitizer corrections:*

name:: Demonstration corrections for setup with two 3458A digitizers

*// names of the channels as they appear in the digitizer identification:*

*// these are exact unique names of the channels in the order that will be loaded to the SW*

**#startmatrix**:: channel identifiers

HP3458A, sn. MY45053095

HP3458A, sn. MY45053104

**#endmatrix**:: channel identifiers

*// relative links to the files with channel corrections for each channel:*

**#startmatrix**:: channel correction paths

..\channel\_MY45053095\HP3458\_MY45053095.info

..\channel\_MY45053104\HP3458\_MY45053104.info

**#endmatrix**:: channel correction paths

*// here follows definitions of ANY correction tables*

*// ..........*

The identifier of the correction type “**type**” must be set to value “digitizer”. The “**name**” is any string describing the correction data file. Next, there is a list of a digitizer channel identifiers “**channel identifiers”.** This is the list of digitizer channel identification strings exactly as they are returned during the instrument identification in the TWM tool. These are used to filter the correction file only for particular instruments. Otherwise the TWM tool will return an error if the processing of data is initiated. Next item is “**channel correction paths”** which are relative paths to the files with the channel corrections, one for each channel of the digitizer. Next, the correction data tables follows.

#### Inter-channel time-shift correction (optional)

The table “**interchannel timeshift**” defines correction values for time shifts between the channels of the digitizer. It must be a row vector of values, one for each channel, that defines correction of time shift of each channel relative to the first channel in the “**channel indentifiers**” list, e.g. for three channels:

**#startsection**:: interchannel timeshift

**#startmatrix**:: value

**0.0; 0.010000; 0.020000**

**#endmatrix**:: value

**#startmatrix**:: uncertainty

**0.0; 0.000012; 0.000011**

**#endmatrix**:: uncertainty

**#endsection**:: interchannel timeshift

Note the first value is always zero. Shown example means second channel correction is (0.010000 ± 0.000012) s, and third correction is (0.020000 ± 0.000011) s. Note it is a correction factor, not a time shift, so the sign of the values is opposite to the measured time shifts. The correction is optional. By default the time shifts and uncertainty is zero.

The values of the time shift are combined with the timestamps coming from the digitizer and are passed to the QWTB algorithm according to the rules defined in [4].

#### Timebase correction (optional)

The correction “**timebase correction**” defines relative correction to the error of timebase of the digitizer. It is optional parameter. E.g.: value +1e-7 means the actual timebase of the digitizer *f\_ref* is: *f\_ref = f\_nom\*(1 + 1e-7)*. Note the value is common for all channels thus it was placed in the digitizer correction instead of channel correction.

The value will be passed to the QWTB under quantity names:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| adc\_freq.v  adc\_freq.u | 0  0 | Value of correction  Abs. std. uncertainty |

#### Inter-channel crosstalk

To be defined.

### Channel corrections

Channel corrections define corrections that apply only to a single channel of the digitizer. Example of the channel correction file header:

*// type of the correction*

type:: channel

*// correction name string*

name:: Channel correction HP3458A, sn. MY45053095

*// device/channel identification as it appears in the digitizer identification*

*// note: leave empty or remove if this correction should be independent of the instrument/channel*

channel identifier:: HP3458A, sn. MY45053095

*// here follows definitions of ANY correction tables*

*// ..........*

The “**type**” must be “channel” for the channel correction file. “**name**” is any string describing the correction file. “**channel identifier**” is optional item that will cause the TWM correction loader will throw and error if this channel correction is applied digitizer channel with different identification. It must be the exact string as returned by the TWM tool during digitizer identification. It may be removed if it is not required.

#### Nominal gain (optional)

Optional correction “**nominal gain**” defines DC gain of the digitizer and its std. uncertainty. The value is combined with relative channel frequency transfer to absolute transfer (see below). Example:

**#startsection**:: nominal gain

**#startmatrix**:: value

**1.000005**

**#endmatrix**:: value

**#startmatrix**:: uncertainty

**0.000003**

**#endmatrix**:: uncertainty

**#endsection**:: nominal gain

#### Gain frequency transfer (optional)

Optional correction “**gain transfer**” defines relative frequency dependence of the gain of the digitizer channel. It is combined with the nominal gain to absolute gain transfer:

abs gain = nominal gain \* gain

abs gain uncertainty = sqrt((nominal gain uncertainty)^2 + u(gain)^2)

The calculated absolute correction value is multiplied by the measured amplitude to get actual amplitude of the input signal.

The correction data is 2D CSV table dependent on the frequency and amplitude. Example of the correction section:

**#startsection**:: gain transfer

**#startmatrix**:: value

**csv\tfer\_gain.csv**

**#endmatrix**:: value

**#endsection**:: gain transfer

2D CSV table format:

|  |  |
| --- | --- |
| x-axis: | harmonic component amplitude [V] |
| y-axis: | harmonic component frequency [Hz] |
| Quantities: | gain – relative gain  u(gain) – absolute std. uncertainty of gain |

The value of absolute gain will be passed to the QWTB under quantity names:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*adc\_gain\_f.v | [] | Frequency axis |
| \*adc\_gain\_a.v | [] | Amplitude axis |
| \*adc\_gain.v  \*adc\_gain.u | 1  0 | Value of correction  Abs. std. uncertainty |

\* - channel prefix [4] (e.g. “u\_”, “i\_”, “u\_lo\_”, …)

#### Phase frequency transfer (optional)

Optional correction “**phase transfer**” defines frequency dependence of the correction to the phase error of the digitizer channel. It is the value which must be added to the measured phase of the harmonic component to get actual phase angle of the input signal.

The correction data is 2D CSV table dependent on the frequency and amplitude. Example of the correction section:

**#startsection**:: phase transfer

**#startmatrix**:: value

**csv\tfer\_phi.csv**

**#endmatrix**:: value

**#endsection**:: phase transfer

2D CSV table format:

|  |  |
| --- | --- |
| x-axis: | harmonic component amplitude [V] |
| y-axis: | harmonic component frequency [Hz] |
| Quantities: | phi – phase correction [rad]  u(phi) – absolute std. uncertainty of gain [rad] |

The value will be passed to the QWTB under quantity names:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*adc\_phi\_f.v | [] | Frequency axis |
| \*adc\_phi\_a.v | [] | Amplitude axis |
| \*adc\_phi.v  \*adc\_phi.u | 0  0 | Value of correction  Abs. std. uncertainty |

\* - channel prefix [4] (e.g. “u\_”, “i\_”, “u\_lo\_”, …)

#### DC offset (optional)

Correction “**dc offset**” defines DC offset of the digitizer and its uncertainty. Note it is DC offset, not the correction! Example of the correction section:

**#startsection**:: dc offset

**#startmatrix**:: value

**1.234e-6**

**#endmatrix**:: value

**#startmatrix**:: uncertainty

**2.5e-6**

**#endmatrix**:: uncertainty

**#endsection**:: dc offset

Non-zero value enables the correction. This correction has no uncertainty value.

The value will be passed to the QWTB under quantity names:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*adc\_offset.v | 0 | DC offset |
| \*adc\_offset.u | 0 | Absolute uncertainty of DC offset |

\* - channel prefix [4] (e.g. “u\_”, “i\_”, “u\_lo\_”, …)

#### Aperture correction (optional)

Correction “**aperture correction**” defines whether the TWM algorithms should perform gain and phase correction to the effect of the aperture time of the ADC. The correction has effect only for digitizers that have aperture parameter such as 3458A. It will perform corrections:

k\_gain = Ta\*pi\*f/sin(Ta\*pi\*f) [-],

k\_phi = Ta\*pi\*f [rad],

where the *Ta* is aperture time from measurement header. Example of the correction section:

**#startsection**:: aperture correction

**#startmatrix**:: value

**1**

**#endmatrix**:: value

**#endsection**:: aperture correction

Non-zero value enables the correction. This correction has no uncertainty value.

The value will be passed to the QWTB under quantity names:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*adc\_aper\_corr.v | 1 | 0/1 to disable/enable correction |

\* - channel prefix [4] (e.g. “u\_”, “i\_”, “u\_lo\_”, …)

#### SFDR value (optional)

Correction “**sfdr**” defines effects of the distortion of the digitizer. It is defined as 2D CSV table of SFDR values dependent on fundamental component amplitude and frequency. It is a value in [dB]. Note the values are positive, i.e.: 120 dB means max spur amplitude is *A0\*10^-(120/20)*. The SFDR value is not correction as such as SFDR cannot be used to correct anything. It is just used by the TWM algorithms to estimate uncertainty caused by the SFDR.

Example of the correction section:

**#startsection**:: sfdr

**#startmatrix**:: value

**csv\sfdr.csv**

**#endmatrix**:: value

**#endsection**:: sfdr

2D CSV table format:

|  |  |
| --- | --- |
| x-axis: | Fundamental harmonic component amplitude [V] |
| y-axis: | Fundamental harmonic component frequency [Hz] |
| Quantities: | sfdr – positive SFDR value [dB] |

The value will be passed to the QWTB under quantity names:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*adc\_sfdr\_f.v | [] | Frequency axis |
| \*adc\_sfdr\_a.v | [] | Amplitude axis |
| \*adc\_sfdr.v | 180 | SFDR values |

\* - channel prefix [4] (e.g. “u\_”, “i\_”, “u\_lo\_”, …)

#### RMS jitter (optional)

Correction “**rms jitter**” defines rms value of the channel time jitter in [s]. Example of the jitter correction section:

**#startsection**:: rms jitter

**#startmatrix**:: value

**1e-8**

**#endmatrix**:: value

**#endsection**:: rms jitter

The value will be passed to the QWTB under quantity names:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*adc\_jitter.v | 0 | RMS jitter value [s] |

\* - channel prefix [4] (e.g. “u\_”, “i\_”, “u\_lo\_”, …)

#### Input admittance (optional)

Correction “**input admittance**” defines input admittance of the digitizer channel. It is used as a part of the transducer loading corrections where it is component “**Yin**” (and “**lo\_Yin**” for differential connection). The correction data are in form of 1D CSV table. Example of the correction section:

**#startsection**:: input admittance

**#startmatrix**:: value

**csv\Y\_inp.csv**

**#endmatrix**:: value

**#endsection**:: input admittance

1D CSV table format:

|  |  |
| --- | --- |
| y-axis: | Frequency [Hz] |
| Quantities: | Cp – parallel capacitance [F]  Gp – parallel loss conductance [S]  u(Cp) – absolute std. uncertainty of Cp [F]  u(Gp) – absolute std. uncertainty of Gp [S] |

The value will be passed to the QWTB under quantity names:

|  |  |  |
| --- | --- | --- |
| *QWTB name* | *Default value* | *Meaning* |
| \*adc\_Yin\_f.v | [] | Frequency axis |
| \*adc\_Yin\_Cp.v  \*adc\_Yin\_Cp.u  \*adc\_Yin\_Gp.v  \*adc\_Yin\_Gp.u | 0  0  1e-12  0 | Cp value [F]  u(Cp) [F]  Gp value [S]  u(Gp) [S] |

\* - channel prefix [4] (e.g. “u\_”, “i\_”, “u\_lo\_”, …)