Calibration Programming Manual

A document describing the commands used to communicate to the instrument via Bluetooth

AVSpekter d.o.o
Slovenia
November 2023

Contents

1 Communication Protocol

The communication is based on the SCPI protocol and it's communicated over Bluetooth. The data contained in the `<command parameter>` and `<config value>` are numeric values of the enums provided in the calibration headers on the export_ag repository.

1.1 Bluetooth configuration

Bluetooth device contains three characteristics:

- `3000` - write
- `5000` - notify

The characteristic `3000` is used for writing commands to the device and the characteristic `5000` is used for asynchronous reading from the device.

1.2 Starting calibration

To start the calibration procedure, a command `ENTER_CALIBRATION` should be sent to the device, so that it enters the `Calibration` mode.

1.3 Commands

1.3.1 Command Structure

The commands are written in the following format:

```
<command name>:<command parameter>[<config value>][?]\r\n
```

1.3.2 Commands

The communication protocol includes the following commands:

- CONFigure
- MEASure
- STOP
- FETCh

In the above commands, lowercase letters can be left out as they are optional.

1.3.3 Query Commands

Query commands are the commands that request data from the device.

We support the following query commands:

- `CONFigure`
- `FETCh`

Query commands have to end with `?`.

1.3.4 Configuration Commands

Configuration commands are the commands that configure the device. These commands can be both query and configuration commands.

Format for configuration is as the following:

```
CONFigure:<command parameter> <config value>\r\n
```

Format for query is as the following:

```
CONFigure:<command parameter>?\r\n
```

1.3.5 Measurement commands

Measurement commands start measuring.

Format is as the following:

```
MEASure:<command parameter>\r\n
```

1.4 Responses

Currently we only support two types of responses: `Enum`s and `float`s.

- Floating point values are always `0` padded to 4 digits before the decimal point and 4 digits after. Decimal point is always a `.` (period).
  - `0000.0000`
  - `1234.5678`
- Enum values are sent over as an integer value.
  - `103`
  - `418`

2 Measurements

If results and settings list enums below them, then the result could be one of these enums.

2.1 Settings

2.1.1 Global settings

- CAL_SETT_PP_LOAD_SETTING
  - CAL_SETT_VAL_PP_LOAD_OPEN
  - CAL_SETT_VAL_PP_LOAD_13A
  - CAL_SETT_VAL_PP_LOAD_20A
  - CAL_SETT_VAL_PP_LOAD_32A
  - CAL_SETT_VAL_PP_LOAD_63_70A
  - CAL_SETT_VAL_PP_LOAD_ERROR
- CAL_SETT_CP_LOAD_SETTING
  - CAL_SETT_VAL_CP_LOAD_A
  - CAL_SETT_VAL_CP_LOAD_B_HIGH
  - CAL_SETT_VAL_CP_LOAD_B_MID
  - CAL_SETT_VAL_CP_LOAD_B_LOW
  - CAL_SETT_VAL_CP_LOAD_C_HIGH
  - CAL_SETT_VAL_CP_LOAD_C_MID
  - CAL_SETT_VAL_CP_LOAD_C_LOW
  - CAL_SETT_VAL_CP_LOAD_D_HIGH
  - CAL_SETT_VAL_CP_LOAD_D_MID
  - CAL_SETT_VAL_CP_LOAD_D_LOW
  - CAL_SETT_VAL_CP_LOAD_PE_OPEN
- CAL_SETT_CP_ERROR
  - CAL_SETT_VAL_OFF
  - CAL_SETT_VAL_CP_ERROR_0R
  - CAL_SETT_VAL_CP_ERROR_80R
  - CAL_SETT_VAL_CP_ERROR_120R
- CAL_SETT_CP_DIODE_SHORT
  - CAL_SETT_VAL_OFF
  - CAL_SETT_VAL_ON
- CAL_SETT_CP_L_BYPASS
  - CAL_SETT_VAL_OFF
  - CAL_SETT_VAL_ON

2.2 CP Voltage

2.2.1 Measurement command

- CAL_MEASUREMENT_CP_VOLTAGE_MEAS

2.2.2 Result commands

- CAL_RES_UCP_P
- CAL_RES_UCP_N
- CAL_RES_F_CP
- CAL_RES_DUTY_CYCLE_CP

2.3 CP Resistance

2.3.1 Measurement command

- CAL_MEASUREMENT_CP_RESISTANCE_MEAS

2.3.2 Result commands

- CAL_RES_CP_RESISTANCE

2.4 Fuse detect

2.4.1 Measurement command

- CAL_MEASUREMENT_FUSE_DET_MEAS

2.4.2 Result commands

- CAL_RES_CP_FUSE
- CAL_RES_PP_FUSE

2.5 PP Resistance

2.5.1 Measurement command

- CAL_MEASUREMENT_PP_RESISTANCE_MEAS

2.5.2 Result commands

- CAL_RES_PP_RESISTANCE
- CAL_RES_PP_VOLTAGE

2.6 PE Pre-Test

2.6.1 Measurement command

- CAL_MEASUREMENT_PE_PRETEST_MEAS

2.6.2 Result commands

- CAL_RES_U_FAULT
- CAL_RES_PE_FAULT_DETECTED

2.7 Battery test

2.7.1 Measurement command

- CAL_MEASUREMENT_BATTERY_MEAS

2.7.2 Result commands

- CAL_RES_BATTERY_VOLTAGE
- CAL_RES_BATTERY_TYPE
  - CAL_RES_VAL_NIMH
  - CAL_RES_VAL_ALKALINE
- CAL_RES_BAT_RESISTANCE

2.8 Mains V

2.8.1 Measurement command

- CAL_MEASUREMENT_MAINS_MEAS

2.8.2 Result commands

- CAL_RES_UL1PE_VOLTAGE
- CAL_RES_UL2PE_VOLTAGE
- CAL_RES_UL3PE_VOLTAGE
- CAL_RES_UNPE_VOLTAGE
- CAL_RES_UL1N_VOLTAGE
- CAL_RES_UL2N_VOLTAGE
- CAL_RES_UL3N_VOLTAGE
- CAL_RES_UL12_VOLTAGE
- CAL_RES_UL23_VOLTAGE
- CAL_RES_UL31_VOLTAGE
- CAL_RES_FREQUENCY
- CAL_RES_3PSEQ
  - CAL_RES_VAL_UNDEF
  - CAL_RES_VAL_LEFT
  - CAL_RES_VAL_RIGHT

2.9 RCD

2.9.1 Measurement command

- CAL_MEASUREMENT_RCD_MEAS

2.9.2 Setting commands

- CAL_SETT_RCD_TEST_SELECTION
  - CAL_SETT_VAL_RCD_TEST_RAMP
  - CAL_SETT_VAL_RCD_TEST_TRIP
  - CAL_SETT_VAL_RCD_TEST_UF
  - CAL_SETT_VAL_RCD_TEST_GFCI
- CAL_SETT_RCD_SHAPE_SELECTION
  - CAL_SETT_VAL_RCD_SHAPE_AC
  - CAL_SETT_VAL_RCD_SHAPE_A
  - CAL_SETT_VAL_RCD_SHAPE_DC
  - CAL_SETT_VAL_RCD_SHAPE_RDC_DD
  - CAL_SETT_VAL_RCD_SHAPE_GFCI_6MA
  - CAL_SETT_VAL_RCD_SHAPE_GFCI_20MA
- CAL_SETT_RCD_MULT_SELECTION
  - CAL_SETT_VAL_RCD_MULT_X05
  - CAL_SETT_VAL_RCD_MULT_X1
  - CAL_SETT_VAL_RCD_MULT_X5
  - CAL_SETT_VAL_RCD_MULT_X10
  - CAL_SETT_VAL_RCD_MULT_X33
- CAL_SETT_RCD_POLARITY_SELECTION
  - CAL_SETT_VAL_RCD_POLARITY_POS
  - CAL_SETT_VAL_RCD_POLARITY_NEG
- CAL_SETT_RCD_VOLTAGE_ONLY
  - CAL_SETT_VAL_OFF
  - CAL_SETT_VAL_ON

2.9.3 Result commands

- CAL_RES_UL1N_VOLTAGE_RCD
- CAL_RES_UNPE_VOLTAGE_RCD
- CAL_RES_UL1PE_VOLTAGE_RCD
- CAL_RES_TIME_RCD
- CAL_RES_CURRENT_RCD
- CAL_RES_UF_VOLTAGE_RCD
- CAL_RES_RE_RCD
- CAL_RES_TEMP_RCD

2.10 PP Voltage

2.10.1 Measurement command

- CAL_MEASUREMENT_PP_VOLTAGE_MEAS

2.10.2 Result commands

- CAL_RES_PP_VOLTAGE