Keithley 2400 SourceMeter

class `pymeasure.instruments.keithley.Keithley2400`(*adapter*, ***kwargs*)

Bases: `pymeasure.instruments.instrument.Instrument`, `pymeasure.instruments.keithley.buffer.KeithleyBuffer`

Represents the Keithely 2400 SourceMeter and provides a high-level interface for interacting with the instrument.

```
keithley = Keithley2400("GPIB::1")

keithley.apply_current()                # Sets up to source current
keithley.source_current_range = 10e-3   # Sets the source current range to 10 mA
keithley.compliance_voltage = 10        # Sets the compliance voltage to 10 V
keithley.source_current = 0             # Sets the source current to 0 mA
keithley.enable_source()                # Enables the source output

keithley.measure_voltage()              # Sets up to measure voltage

keithley.ramp_to_current(5e-3)          # Ramps the current to 5 mA
print(keithley.voltage)                 # Prints the voltage in Volts

keithley.shutdown()                     # Ramps the current to 0 mA and disables output
```

apply_current(*current_range=None*, *compliance_voltage=0.1*)

Configures the instrument to apply a source current, and uses an auto range unless a current range is specified. The compliance voltage is also set.

Parameters:
- **compliance_voltage** – A float in the correct range for a `compliance_voltage`
- **current_range** – A `current_range` value or None

apply_voltage(*voltage_range=None*, *compliance_current=0.1*)

Configures the instrument to apply a source voltage, and uses an auto range unless a voltage range is specified. The compliance current is also set.

Parameters:
- **compliance_current** – A float in the correct range for a `compliance_current`
- **voltage_range** – A `voltage_range` value or None

property `auto_output_off`

A boolean property that enables or disables the auto output-off. Valid values are True (output off after measurement) and False (output stays on after measurement).

`auto_range_source`()

Configures the source to use an automatic range.

property `auto_zero`

A property that controls the auto zero option. Valid values are True (enabled) and False (disabled) and 'ONCE' (force immediate).

`beep`(*frequency*, *duration*)

Sounds a system beep.

Parameters:

- **frequency** – A frequency in Hz between 65 Hz and 2 MHz
- **duration** – A time in seconds between 0 and 7.9 seconds

`binary_values`(*command*, *query_delay=0*, ***kwargs*)

Returns a numpy array from a query for binary data.

Parameters:

- **command** – Command to be sent to the instrument.
- **query_delay** – Delay between writing and reading in seconds.
- **kwargs** – Arguments for `Adapter.read_binary_values()`.

Returns: NumPy array of values

property `buffer_data`

Returns a numpy array of values from the buffer.

property `buffer_points`

An integer property that controls the number of buffer points. This does not represent actual points in the buffer, but the configuration value instead.

`check_errors`()

Logs any system errors reported by the instrument.

property `complete`

This property allows synchronization between a controller and a device. The Operation Complete query places an ASCII character 1 into the device’s Output Queue when all pending selected device operations have been finished.

property `compliance_current`

A floating point property that controls the compliance current in Amps.

property `compliance_voltage`

A floating point property that controls the compliance voltage in Volts.

`config_buffer`(*points=64*, *delay=0*)

Configures the measurement buffer for a number of points, to be taken with a specified delay.

Parameters:

- **points** – The number of points in the buffer.
- **delay** – The delay time in seconds.

property `current`

Reads the current in Amps, if configured for this reading.

property `current_nplc`

A floating point property that controls the number of power line cycles (NPLC) for the DC current measurements, which sets the integration period and measurement speed. Takes values from 0.01 to 10, where 0.1, 1, and 10 are Fast, Medium, and Slow respectively.

property `current_range`

A floating point property that controls the measurement current range in Amps, which can take values between -1.05 and +1.05 A. Auto-range is disabled when this property is set.

`disable_buffer`()

Disables the connection between measurements and the buffer, but does not abort the measurement process.

`disable_output_trigger`()

Disables the output trigger for the Trigger layer

`disable_source()`

Disables the source of current or voltage depending on the configuration of the instrument.

property `display_enabled`

A boolean property that controls whether or not the display of the sourcemeter is enabled. Valid values are True and False.

`enable_source()`

Enables the source of current or voltage depending on the configuration of the instrument.

property `error`

Returns a tuple of an error code and message from a single error.

property `filter_count`

A integer property that controls the number of readings that are acquired and stored in the filter buffer for the averaging

property `filter_state`

A string property that controls if the filter is active.

property `filter_type`

A String property that controls the filter's type. REP : Repeating filter MOV : Moving filter

property `id`

Requests and returns the identification of the instrument.

`is_buffer_full()`

Returns True if the buffer is full of measurements.

property **`line_frequency`**

An integer property that controls the line frequency in Hertz. Valid values are 50 and 60.

property **`line_frequency_auto`**

A boolean property that enables or disables auto line frequency. Valid values are True and False.

property **`max_current`**

Returns the maximum current from the buffer

property **`max_resistance`**

Returns the maximum resistance from the buffer

property **`max_voltage`**

Returns the maximum voltage from the buffer

property **`maximums`**

Returns the calculated maximums for voltage, current, and resistance from the buffer data as a list.

property **`mean_current`**

Returns the mean current from the buffer

property **`mean_resistance`**

Returns the mean resistance from the buffer

property **`mean_voltage`**

Returns the mean voltage from the buffer *property* `means`

Reads the calculated means (averages) for voltage, current, and resistance from the buffer data as a list.

property `measure_concurent_functions`

A boolean property that enables or disables the ability to measure more than one function simultaneously. When disabled, volts function is enabled. Valid values are True and False.

`measure_current`(*nplc=1*, *current=0.000105*, *auto_range=True*)

Configures the measurement of current.

Parameters:

- **nplc** – Number of power line cycles (NPLC) from 0.01 to 10
- **current** – Upper limit of current in Amps, from -1.05 A to 1.05 A
- **auto_range** – Enables auto_range if True, else uses the set current

`measure_resistance`(*nplc=1*, *resistance=210000.0*, *auto_range=True*)

Configures the measurement of resistance.

Parameters:

- **nplc** – Number of power line cycles (NPLC) from 0.01 to 10
- **resistance** – Upper limit of resistance in Ohms, from -210 MOhms to 210 MOhms
- **auto_range** – Enables auto_range if True, else uses the set resistance

`measure_voltage`(*nplc=1*, *voltage=21.0*, *auto_range=True*)

Configures the measurement of voltage.

Parameters:

- **nplc** – Number of power line cycles (NPLC) from 0.01 to 10
- **voltage** – Upper limit of voltage in Volts, from -210 V to 210 V
- **auto_range** – Enables auto_range if True, else uses the set voltage

property `min_current`

Returns the minimum current from the buffer

property `min_resistance`

Returns the minimum resistance from the buffer

property `min_voltage`

Returns the minimum voltage from the buffer

property `minimums`

Returns the calculated minimums for voltage, current, and resistance from the buffer data as a list.

property `options`

Requests and returns the device options installed.

property `output_off_state`

Select the output-off state of the SourceMeter. HIMP : output relay is open, disconnects external circuitry. NORM : V-Source is selected and set to 0V, Compliance is set to 0.5% full scale of the present current range. ZERO : V-Source is selected and set to 0V, compliance is set to the programmed Source I value or to 0.5% full scale of the present current range, whichever is greater. GUAR : I-Source is selected and set to 0A

`output_trigger_on_external`(*line=1*, *after='DEL'*)

Configures the output trigger on the specified trigger link line number, with the option of supplying the part of the measurement after which the trigger should be generated (default to delay, which is right before the measurement)

Parameters:

- **line** – A trigger line from 1 to 4
- **after** – An event string that determines when to trigger

`ramp_to_current`(*target_current*, *steps=30*, *pause=0.02*)

Ramps to a target current from the set current value over a certain number of linear steps, each separated by a pause duration.

Parameters:

- **target_current** – A current in Amps
- **steps** – An integer number of steps

- **pause** – A pause duration in seconds to wait between steps

`ramp_to_voltage`(*target_voltage*, *steps=30*, *pause=0.02*)

Ramps to a target voltage from the set voltage value over a certain number of linear steps, each separated by a pause duration.

Parameters:

- **target_voltage** – A voltage in Amps
- **steps** – An integer number of steps
- **pause** – A pause duration in seconds to wait between steps

`read_bytes`(*count*, ***kwargs*)

Read a certain number of bytes from the instrument.

Parameters:

- **count** (*int*) – Number of bytes to read. A value of -1 indicates to read the whole read buffer.
- **kwargs** – Keyword arguments for the adapter.

Returns bytes: Bytes response of the instrument (including termination).

`reset`()

Resets the instrument and clears the queue.

`reset_buffer`()

Resets the buffer.

property `resistance`

Reads the resistance in Ohms, if configured for this reading.

property `resistance_nplc`

A floating point property that controls the number of power line cycles (NPLC) for the 2-wire resistance measurements, which sets the integration period and measurement speed. Takes values from 0.01 to 10, where 0.1, 1, and 10 are Fast, Medium, and Slow respectively.

property `resistance_range`

A floating point property that controls the resistance range in Ohms, which can take values from 0 to 210 MOhms. Auto-range is disabled when this property is set.

`sample_continuously`()

Causes the instrument to continuously read samples and turns off any buffer or output triggering

`set_timed_arm`(*interval*)

Sets up the measurement to be taken with the internal trigger at a variable sampling rate defined by the interval in seconds between sampling points

`set_trigger_counts`(*arm*, *trigger*)

Sets the number of counts for both the sweeps (arm) and the points in those sweeps (trigger), where the total number of points can not exceed 2500

`shutdown`()

Ensures that the current or voltage is turned to zero and disables the output.

property **`source_current`**

A floating point property that controls the source current in Amps.

property **`source_current_range`**

A floating point property that controls the source current range in Amps, which can take values between -1.05 and +1.05 A. Auto-range is disabled when this property is set.

property **`source_delay`**

A floating point property that sets a manual delay for the source after the output is turned on before a measurement is taken. When this property is set, the auto delay is turned off. Valid values are between 0 [seconds] and 999.9999 [seconds].

property **`source_delay_auto`**

A boolean property that enables or disables auto delay. Valid values are True and False.

property **`source_enabled`**

A boolean property that controls whether the source is enabled, takes values True or False. The convenience methods `enable_source()` and `disable_source()` can also be used.

property **`source_mode`**

A string property that controls the source mode, which can take the values 'current' or 'voltage'. The convenience methods `apply_current()` and `apply_voltage()` can also be used.

property **`source_voltage`**

A floating point property that controls the source voltage in Volts.

property **`source_voltage_range`**

A floating point property that controls the source voltage range in Volts, which can take values from -210 to 210 V. Auto-range is disabled when this property is set.

property **`standard_devs`**

Returns the calculated standard deviations for voltage, current, and resistance from the buffer data as a list.

`start_buffer`()

Starts the buffer.

`status`()

Requests and returns the status byte and Master Summary Status bit.

property **`std_current`**

Returns the current standard deviation from the buffer

property **`std_resistance`**

Returns the resistance standard deviation from the buffer

property **`std_voltage`**

Returns the voltage standard deviation from the buffer

`stop_buffer`()

Aborts the buffering measurement, by stopping the measurement arming and triggering sequence. If possible, a Selected Device Clear (SDC) is used.

`triad`(*base_frequency*, *duration*)

Sounds a musical triad using the system beep.

Parameters:

- **base_frequency** – A frequency in Hz between 65 Hz and 1.3 MHz
- **duration** – A time in seconds between 0 and 7.9 seconds

`trigger()`

Executes a bus trigger, which can be used when `trigger_on_bus()` is configured.

property `trigger_count`

An integer property that controls the trigger count, which can take values from 1 to 9,999.

property `trigger_delay`

A floating point property that controls the trigger delay in seconds, which can take values from 0 to 999.9999 s.

`trigger_immediately()`

Configures measurements to be taken with the internal trigger at the maximum sampling rate.

`trigger_on_bus()`

Configures the trigger to detect events based on the bus trigger, which can be activated by `trigger()`.

`trigger_on_external`(*line=1*)

Configures the measurement trigger to be taken from a specific line of an external trigger

Parameters: **line** – A trigger line from 1 to 4

`use_front_terminals()`

Enables the front terminals for measurement, and disables the rear terminals.

`use_rear_terminals()`

Enables the rear terminals for measurement, and disables the front terminals.

property `voltage`

Reads the voltage in Volts, if configured for this reading.

property `voltage_nplc`

A floating point property that controls the number of power line cycles (NPLC) for the DC voltage measurements, which sets the integration period and measurement speed. Takes values from 0.01 to 10, where 0.1, 1, and 10 are Fast, Medium, and Slow respectively.

property `voltage_range`

A floating point property that controls the measurement voltage range in Volts, which can take values from -210 to 210 V. Auto-range is disabled when this property is set.

`wait_for`(*query_delay=0*)

Wait for some time. Used by 'ask' to wait before reading.

Parameters: **query_delay** – Delay between writing and reading in seconds.

`wait_for_buffer`(*should_stop=<function KeithleyBuffer.<lambda>>*, *timeout=60*, *interval=0.1*)

Blocks the program, waiting for a full buffer. This function returns early if the `should_stop` function returns True or the timeout is reached before the buffer is full.

Parameters:

- **should_stop** – A function that returns True when this function should return early
- **timeout** – A time in seconds after which this function should return early
- **interval** – A time in seconds for how often to check if the buffer is full

property `wires`

An integer property that controls the number of wires in use for resistance measurements, which can take the value of 2 or 4.

`write_binary_values`(*command*, *values*, **args*, ***kwargs*)

Write binary values to the device.

Parameters:

- **command** – Command to send.
- **values** – The values to transmit.
- ****kwargs** (**args,*) – Further arguments to hand to the Adapter.

`write_bytes`(*content*, ***kwargs*)

Write the bytes *content* to the instrument.