Version: 1.0

EM1

EM1 component handles the data collection and processing from energy meter devices like the `ShellyProEM`.

- `EM1.SetConfig` to update the component's configuration
- `EM1.GetConfig` to obtain the component's configuration
- `EM1.GetStatus` to obtain the component's status
- `EM1.CalibrateFrom` calibrate a phase CT from another phase's CT (if applicable)
- `EM1.RevertToFactoryCalibration` reset a **user** calibrated EM1 component to factory defaults (if applicable)
- `EM1.GetCTTypes` to obtain list of supported current transformer types

Methods

EM1.SetConfig

Properties:

Property	Type	Description
`id`	*number*	Id of the EM component instance
`config`	*object*	Configuration that the method takes

Find more about the config properties in config section

EM1.GetConfig

Properties:

Property	Type	Description
`id`	*number*	Id of the EM1 component instance

Find the EM1.GetConfig response properties in config section

EM1.GetStatus

Properties:

Property	Type	Description
`id`	*number*	Id of the EM1 component instance

Find more about the status response properties in status section

EM1.CalibrateFrom

This method calibrates(aligns the measurements of) an instance of EM1 component to another EM1 component (if applicable).

Request

Parameters:

Property	Type	Description
`id`	*number*	Id of the EM1 component instance which is going to be calibrated
`other_id`	*number*	Id of the EM1 component from witch the calibration data is taken

Response:

`restart_required:true` on success; error if request can't be executed or failed

INFO

The minimum power allowed for the calibration to take place is `500W`. The method takes around 5 seconds to complete, and the response is delayed with the request answer - `restart_required: true` in case of success or an error message in case of `fail`. The reasons for the calibration to `fail` may be - deviation in measurement after calibration on the `from` and `to` EM1 components that indicate incorrect CTs or other problems.

EM1.RevertToFactoryCalibration

This method resets a user-calibrated EM1 component to its factory defaults (if applicable).

Request

Parameters:

Property	Type	Description
`id`	*number*	Id of the EM1 component instance to reset to factory defaults

Response:

`restart_required:true` on success; error if request can't be executed or failed

EM1.GetCTTypes

Properties:

Property	Type	Description
`id`	*number*	Id of the EM1 component instance

Response:

Property	Type	Description
`supported`	*array*	Array of strings of all supported CT types

Configuration

The configuration of the EM1 component shows the energy metering device's name. To Get/Set the configuration of the EM1 component its `id` must be specified.

Properties:

Property	Type	Description
`id`	*number*	Id of the EM1 component instance
`name`	*string or null*	Name of the EM1 instance
`reverse`	*bool*	Reverse CT measurement direction of active power and energy for the EM1 component. *setting the reverse option requires restart*
`ct_type`	*string*	Select the type of Shelly current transformer attached to the device.

> **NOTE**
>
> If `ct_type` is not set, an error `ct_type_not_set` is present in component status.
>
> Supported `ct_type`s can be obtained with EM1.GetCTTypes.

Webhook events

EM1 component supports conditional webhooks.

Events related to the EM1 component that can trigger webhooks:

- `voltage_change` - when the voltage has changed with at least 1V and 5% from the last reported value.
  - `voltage_change` event supports attributes, that can be used to compose conditional webhooks:

Property	Type	Description
`voltage`	*number*	New voltage in Volts

- `current_change` - when the current has changed with at least 0.05A and 5% from the last reported value.
  - `current_change` event supports attributes, that can be used to compose conditional webhooks:

Property	Type	Description
`current`	*number*	New current in Amps

- `active_power_change` - when the active power has changed with at least 10W and 5% from the last reported value.
  - `active_power_change` event supports attributes, that can be used to compose conditional webhooks:

Property	Type	Description
`act_power`	*number*	New active power in Watts

Status

The status of the EM1 Component contains information from the current measurements. To obtain information on the component's status its `id` must be specified. The status contains information for:

- The momentary values of the `current` in `A` (Amps), `voltage` in `V` (Volts), `act_power` in `W` (Watts), `aprt_power` in `VA` (Volt-Ampere), `pf` - power factor is dimensionless, `freq` -

network frequency in `Hz`.

Property	Type	Description
`id`	*number*	Id of the EM1 component instance
`current`	*number or null*	Current measurement value, [A]
`voltage`	*number or null*	Voltage measurement value, [V]
`act_power`	*number or null*	Active power measurement value, [W]
`aprt_power`	*number or null*	Apparent power measurement value, [VA] (if applicable)
`pf`	*number or null*	Power factor measurement value (if applicable)
`freq`	*number or null*	Network frequency measurement value (if applicable)
`calibration`	*string*	Indicates `factory` calibration or which EM1:*id* is the source for calibration
`errors`	*array of type string*	EM1 component error conditions. May contain `power_meter_failure`, `out_of_range:act_power`, `out_of_range:aprt_power`, `out_of_range:voltage`, `out_of_range:current` or `ct_type_not_set`. Present in status only if not empty.

Property	Type	Description
`flags`	*array of type string*	Communicates present conditions, shown if at least one flag is set. Depending on component capabilites may contain: `count_disabled`

> **NOTE**
>
> `count_disabled` is shown when CT types are mixed, showing the blinking light output called `Count` is disabled.

Notifications

StatusChange

Statuschange notification on a regular interval and an error condition change.

Modbus registers

Address	Type	Description
32000	uint32	Timestamp of the last update
32002	boolean	EM1 error
32003	float	Voltage, V
32005	float	Current, A
32007	float	Active power, W
32009	float	Apparent power, VA
32011	float	Power factor

Address	Type	Description
32013	boolean	Overpower error
32014	boolean	Overvoltage error
32015	boolean	Overcurrent error
32016	float	Frequency, Hz
32018		2 registers reserved

If there are more than one EM1 component on the device, every next component's register block start address is calculated by adding 20 to the start address of the previous component. For example, EM1:1 will start at 32020.

Examples

EM1.SetConfig example

Example:

EM1.SetConfig HTTP GET Request | EM1.SetConfig Curl Request | EM1.SetConfig Mos Request

```
http://192.168.33.1/rpc/EM1.SetConfig?id=0&config=
{"name":"abc"}&reverse=true&ct_type="50A"
```

Response

```
{
  "restart_required": false
}
```

EM1.GetConfig example

Example:

EM1.GetConfig HTTP GET Request | EM1.GetConfig Curl Request | EM1.GetConfig Mos Request

```
http://192.168.33.1/rpc/EM1.GetConfig?id=0
```

Response

```
{
  "id": 0,
  "name": null,
  "reverse": false,
  "ct_type": "50A"
}
```

EM1.GetStatus example

EM1.GetStatus HTTP GET Request | EM1.GetStatus Curl Request | EM1.GetStatus Mos Request

```
http://192.168.33.1/rpc/EM1.GetStatus?id=0
```

Response

```
{
  "id": 0,
  "voltage": 236.1,
  "current": 4.029,
  "act_power": 951.2,
  "aprt_power": 951.9,
  "pf": 1,
  "freq": 50,
  "calibration": "factory",
  "errors": [
    "out_of_range:current"
  ],
  "flags": [
    "count_disabled"
```

```
  ]
}
```

EM1.CalibrateFrom example

EM1.CalibrateFrom HTTP GET Request | EM1.CalibrateFrom Curl Request | EM1.CalibrateFrom Mos Request

```
http://192.168.33.1/rpc/EM1.CalibrateFrom?id=0&other_id=1
```

Response

```
{
  "restart_required": true
}
```

EM1.RevertToFactoryCalibration example

EM1.RevertToFactoryCalibration HTTP GET Request | EM1.RevertToFactoryCalibration Curl Request | EM1.RevertToFa Mos Request

```
http://192.168.33.1/rpc/EM1.RevertToFactoryCalibration?id=0
```

Response

```
{
  "restart_required": true
}
```

EM1.GetCTTypes example

Example:

EM1.GetCTTypes HTTP GET Request | **EM1.GetCTTypes Curl Request** | **EM1.GetCTTypes Mos Request**

```
http://192.168.33.1/rpc/EM1.GetCTTypes?id=0
```

Response

```
{
  "supported": [
    "50A"
  ]
}
```