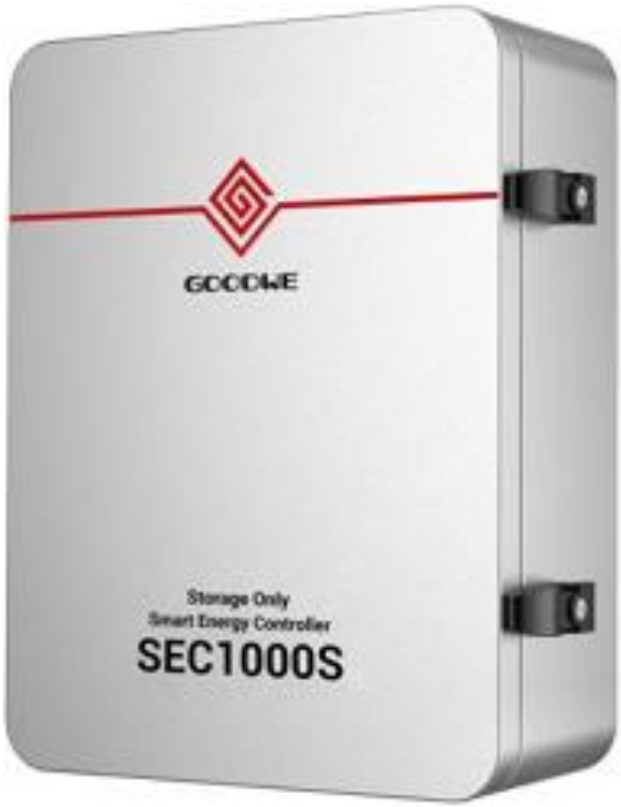


SEC1000/S Integration for Home Assistant

Integration Version Guide: 0.1.1

ABOUT THE HA INTEGRATION SEC1000/S

The integration SEC1000/S provides a set of methods that can be used to set conditions and limits for control of power export to the grid, synchronize time in the device, and read telemetry data that the device provides. The goal of the project is to automate the export of power to the grid, transfer telemetry data from SEMS Portal to Home Assistant, which allows to see the data immediately, and for cybersecurity reasons, cut off the device from the Internet and control it exclusively through its own environment, in which sensitive telemetry data is stored and processed.

Jozef Moravčík
email: jozef.moravcik@moravcik.eu

Content

1. Introduction 2

2. Installation 2

1.1. Step 1 – copying integration files to HA 2

1.2. Step 2 – installation and setup 3

3. Services 7

2.1. Export Disable (export_disable) 7

2.2. Export Enable (export_enable) 7

2.3. Set export limit (set_export_limit) 8

2.4. Get export limit (get_export_limit) 9

2.5. Set Date and Time (set_datetime) 9

2.6. Reset Export Watchdog (reset_export_watchdog) 9

2.7. sec1000_get_telemetry_data 9

4. Sensors 10

5. Visualization 12

5.1. Buttons 12

6. Conclusion 13

1. Introduction

The use of this integration is at your own risk and any potential damage caused by the integration is on the responsibility of the user.

The author created this project with the goal to help SEC1000/S users automate power export control to the grig and other processes used telemetry data and as a part of Home Assistant Open Source concept author doesn't allow the sale of this product for financial profit.

This project was created by a non-professional author, therefore support for this integration is not claimable and is provided only within the author's time capacity.

The current version is usable and precisely tested, available in HACS. In future versions, additional methods will be added, such as improved feedback from services execution, enforcing SEC1000S and inverters to restart export limit mechanism causing temporary freezing of grid export control on the inverter side, etc.

2. Installation

The module is developed as HACS custom integration

Manual installation requires copying the module files into the Home Assistant directory structure, into the "**custom_components**" directory, and the procedure is described in the following steps

1.1. Step 1 – copying integration files to HA

The integration is zipped in a file **goodwe_sec1000_v_0_1_1.ZIP** and contains this structure:

Examples
- └ button - Export ON-OFF.yaml
- └ view - examples.yaml

goodwe_sec1000
- └ **images**
  - └ sec1000s.png
- └ **translations**
  - └ en.json
  - └ sk.json
- └ __init__.py
- └ config_flow.py

└ const.py
└ goodwe_sec1000.py
└ manifest.json
└ sensor.py
└ services.py
└ string.json

lovelace
└ AtomicObjects
└ pv
└ ui-template-pv_set_export_limit_button_style_1.yaml
└ ui-template-pv_get_export_limit_button_style_1.yaml
└ ui-template-pv_set_datetime_button_style_1.yaml
└ ui-template-pv_reset_export_watchdog_button_style_1.yaml

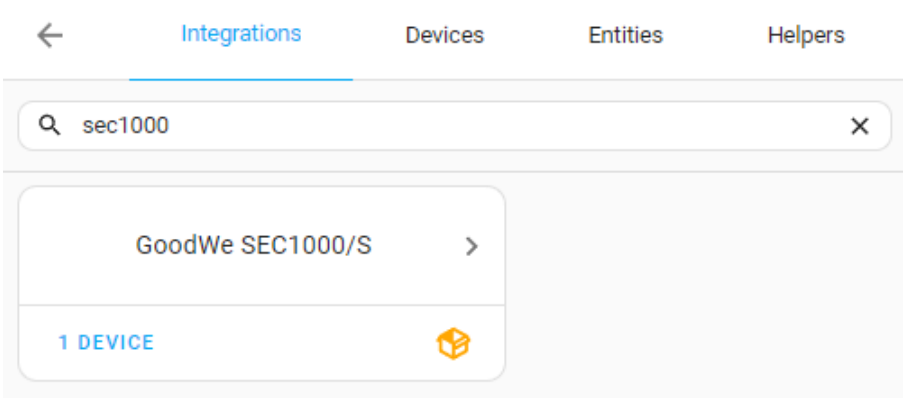
packages
└ **GoodWe**
└ sec1000.yaml

- Unzip the file and copy the folder **goodwe_sec1000** to your Home Assistant folder structure, to target folder: **/config/custom_components/** or /homeassitant/custom_components/
- The current version requires manual integration of script, automation and variables (input_boolean, input_number), all of them you can find in a file packages/GoodWe/**sec1000.yaml**
  - There are various methods how you can integrate the content of sec1000.yaml file into HA. If you are using packages folder for your yaml files, I recommend to copy this file to the folder **packages** in HA.
    - in this case you have to have defined this resource in the **configuration.yaml** file by this definition:

```
homeassistant:
  packages: !include_dir_named packages
```

1.2. Step 2 – installation and setup

- **Installation**
  - After copying all files to HA, or installing from HACS **Restart Home Assistant**
  - Go to **Settings** menu, than select **Devices & services** and click a button **+ ADD INTEGRATION**
  - Find **GoodWe SEC1000/S** Integration


Integrations
Devices
Entities
Helpers
sec1000
GoodWe SEC1000/S
1 DEVICE

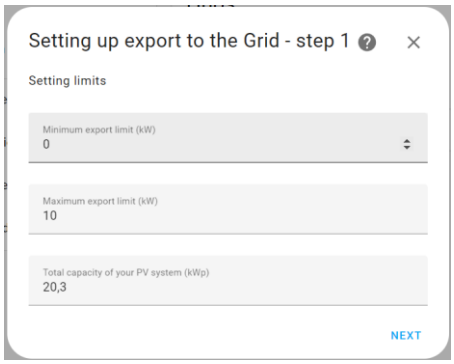
Setup

- **In the first window, set the IP address of your device. According to the GoodWe installation guide for SEC1000, the default IP address should be 192.168.1.200**

  - Set data refresh interval, default is 30 second. I don't recommend to set the refresh interval under 5 second, if the interval is too short, the device can stop to communicate or starts to return fake packets.
    If this happens, increase the time and restart the SEC1000 device (hardware) to resolve the communication failure.

- **In the second window set up export power limits**

  - For simplifying automations two methods (services) are implemented, **Export Disable** and **Export Enable**.
  - When **Export Disable** is called, a value **Minimum export limit** is sent to SEC1000.
  - When **Export Enable** is called, a value **Maximum export limit** is sent to SEC1000.
  - The value **Total capacity of your PV** system is required by the device while a power limit is setup. This value may seem like just a statistical figure, but SEC1000 uses this figure in its power export limitation management algorithm.
  - All values are in kW units defined.

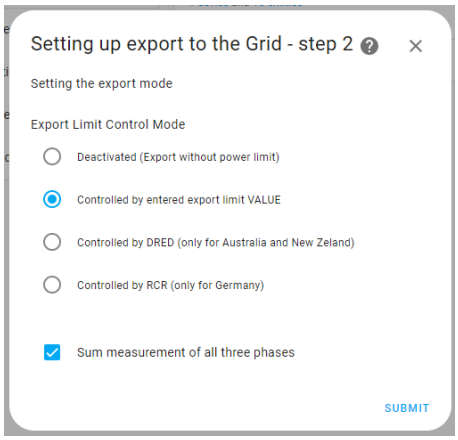


- In the last configuration window set up **Export Limit Control Mode** which is for Central Europe **Controlled by entered export limit VALUE**

  - This Value is defined in the previous configuration window and is automatically sent to the device by calling methods **Export Disable** or **Export Enable**

  - This value also can be sent to the device by calling a method **Set export limit** where the value is as a parameter.

- The value range is from 0 to 100 kW with a resolution of two decimal places
- For Germany use an option **Controlled by RCR**
- For Australia and New Zeland use an option **Controlled by DRED**
- If you want to deactivate Power Limit, use an option **Deactivated**.
  Be aware that deactivating the power limitation will cause the full photovoltaic energy to go to the grid immediatelly!
- A checkbox **Sum measurement of all three phases** defines the way how SEC1000 measures the power flow into the grid. There are only two ways, either the power is measured as a sum on all three phases, or each phase is measured separately (for more details see the GoodWe documentation)



When setting any parameters, you have to be aware of your responsibility for the correct settings and must be fully aware of the consequences of incorrect settings, which may result in possible penalties and other sanctions from the network distributor or authorities.

3. Services

In this version 7 services (methods) are available.

2.1. Export Disable (export_disable)

As mentioned in section 1.2, this method was created for easy control of the export limit, whether in automations or elements such as buttons, etc.
The value sent to the device is taken from configuration, from **Minimum export limit** value which is ussualy 0. Although this value is usually zero, it is configurable because in some special cases it is required to set it to a different, small value, close to zero but not zero.
This service sends also two parameters, **Export Limit Control Mode** and **Sum measurement of all three phases** described in the section 1.2

An example how to call the method by a simple button:

```
- type: button
  show_name: true
  show_icon: true
  name: Export OFF
  icon: mdi:transmission-tower-off
  tap_action:
    action: call-service
    service: goodwe_sec1000.export_disable
```

2.2. Export Enable (export_enable)

Like the previous service , also the service **export_enable** was created for easy control of the export limit. The value sent to the device is taken from configuration, from **Maximum export limit** value. This service sends also two parameters, **Export Limit Control Mode** and **Sum measurement of all three phases** described in the section 1.2

An example how to call the method by a simple button:

```
- type: button
  show_name: true
  show_icon: true
  name: Export ON
  icon: mdi:transmission-tower-export
  tap_action:
    action: call-service
    service: goodwe_sec1000.export_enable
```

2.3. Set export limit (set_export_limit)

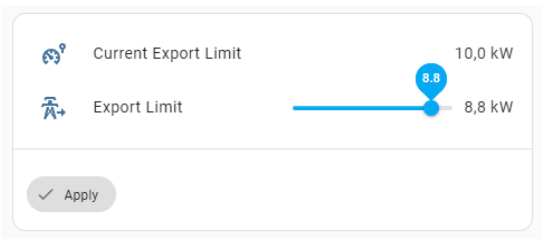
This service sends any export limit value to the device, in the range from 0 to 100kW with a resolution of two decimal places. The value sent to the device is independent of the limit settings described in the section 1.2. However, when setting the export limit value, it sends two settings, namely Export Limit Control Mode and Sum measurement of all three phases.

An example 1, how to call the method with value 7kW by a simple button:

```
- type: button
  show_name: true
  show_icon: true
  name: 7.0 kW
  icon: mdi:transmission-tower-off
  tap_action:
    action: call-service
    service: goodwe_sec1000.set_export_limit
    service_data:
      limit: 7
  grid_options:
    columns: 3
    rows: 2
  icon_height: 50px
```

An example 2, how to call the method by a slider:

```
- type: vertical-stack
  cards:
    - type: entities
      entities:
        - entity: sensor.sec1000_export_limit
          name: Current Export Limit
        - entity: input_number.export_limit
          icon: mdi:transmission-tower-export
          name: Export Limit
          tap_action:
            action: call-service
            service: script.set_export_limit
            data: {}
      footer:
        type: buttons
        entities:
          - entity: script.set_export_limit
            icon: mdi:check
            name: Apply
```



2.4. Get export limit (get_export_limit)

This service reads the current **Export Limit** to a sensor and reads **Control Mode Values** from the device and updates the integration settings.

2.5. Set Date and Time (set_datetime)

This method is intended to set the time and date on the device, with the current time taken from the Home Assistant system. I recommend creating an automation that sets the time on the device once a day, ensuring time consistency between HA and the device.

2.6. Reset Export Watchdog (reset_export_watchdog)

This method is an experimental method that solves the problem of power limitation management (in this integration it is called as the Watchdog) that appears on some inverters. The problem is that when the load increases sharply above the export limit to the grid (for example by turning on a water heating coil or other powerful appliance), the inverter significantly decrease the power drawn from the panels for a few minutes (2-3 minutes), and takes all the power from the battery. After a few minutes, the inverter gradually starts to draw power from the panels, but this behavior is an unnecessarily large load for the battery, shortens the battery lifetime and disrupts the charging process.

This method, when called, completely disables the power limit for 5 seconds and after this time has elapsed, sets everything to its original state, restarting the power limit control algorithm in the inverter and the inverter immediately starts taking full power from the panels.

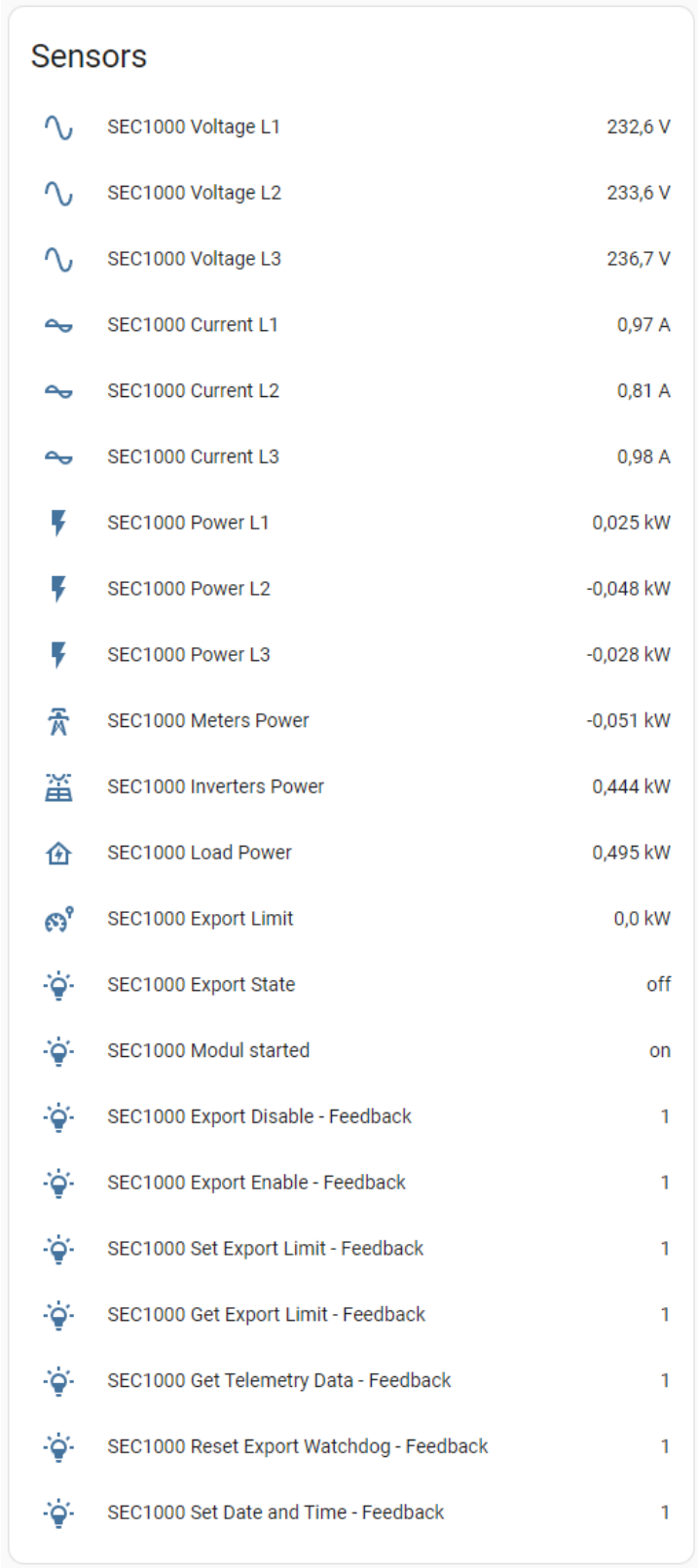
Since this method disables the power limit, **I highly recommend having the process under control and using additional mechanisms to control the power limit**, because if SEC1000 stops communicating during those 5 seconds and the power limit cannot be returned back, the system will export full power to the grid, which may result in exceeding the reserved capacity by the grid operator and possible sanctions.

Therefore, the use of this method is at your own risk and requires that the process be under full control. No other risks have been identified from using this method (such as damage to equipment, etc.)

2.7. sec1000_get_telemetry_data

This method is not accessible by user and is called automatically by timer following **refresh interval value** described in the section 1.2 The timer is automatically launched once the operating system of HA and integration is loaded. All read values are regulary saved in to set of sensors.

4. Sensors


Sensors
SEC1000 Voltage L1 232,6 V
SEC1000 Voltage L2 233,6 V
SEC1000 Voltage L3 236,7 V
SEC1000 Current L1 0,97 A
SEC1000 Current L2 0,81 A
SEC1000 Current L3 0,98 A
SEC1000 Power L1 0,025 kW
SEC1000 Power L2 -0,048 kW
SEC1000 Power L3 -0,028 kW
SEC1000 Meters Power -0,051 kW
SEC1000 Inverters Power 0,444 kW
SEC1000 Load Power 0,495 kW
SEC1000 Export Limit 0,0 kW
SEC1000 Export State off
SEC1000 Modul started on
SEC1000 Export Disable - Feedback 1
SEC1000 Export Enable - Feedback 1
SEC1000 Set Export Limit - Feedback 1
SEC1000 Get Export Limit - Feedback 1
SEC1000 Get Telemetry Data - Feedback 1
SEC1000 Reset Export Watchdog - Feedback 1
SEC1000 Set Date and Time - Feedback 1

In addition to sensors providing telemetry data, the integration also includes these control sensors:

- **SEC1000 Export State:**

  export enable indicator, which has a state = "off" if the export limit value is <= the minimum export limit set in the configuration. If the export limit value is greater than the minimum export limit value, then the state of this sensor is "on"

- **SEC1000 Modul started:**
  when HA is restarted, this sensor indicates that the integration SEC 1000/S is fully ready to operation

- **SEC1000 Export Disable – Feedback**
  SEC1000 Export Enable – Feedback
  SEC1000 Set Export Limit – Feedback
  SEC1000 Get Export Limit – Feedback
  SEC1000 Get Telemetry Data – Feedback
  SEC1000 Reset Export Watchdog – Feedback
  SEC1000 Set Data and Time – Feedback

  All these sensors provide feedback from the progress of communication with the device, which allows the implementation of additional control, protection mechanisms and visual effects of control elements, such as buttons.

  The values of these sensors are a bitmap value and have the same structure.

Feedback value	Error	Description
0	waiting	waiting for response from device
1	OK	communication with the device went well
2	general error	usually it is a data processing error
4	unknown error	For a deeper analysis, the log in HA should be analyzed
8	timeout	The device did not respond in the expected time.
16	unexpected returned packet	The device cannot process a large number of packets at once and with a very short interval between packets the device starts sending unexpected responses or stops responding completely. In this case, the device needs to be restarted.
32	CRC error	
64	socket error	communication error, for example, interrupted connection to SEC1000
128	invalid input parameter	if a method has input parameters (e.g. set export limit) and is supplied with a bad parameter (export limit out of a range), the method can correct the parameter within the set limits, or ignore the command completely and return this error

5. Visualization

This integration also includes examples of visualizing elements using feedback sensors, so that the visualization provides feedback to the user and the control elements reflect the current state.

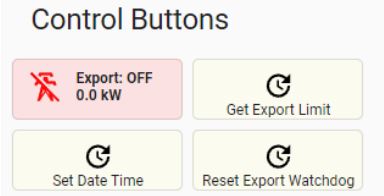
Naturally, everyone can adjust the design to their liking, the examples provided are mainly focused on functionality.

5.1. Buttons

- For implementation these button visualisations is necessary to have **button-card** from HACS installed
- The yaml file for the button implementation can be found in the installation file here:

lovelace
└ AtomicObjects
└ pv
└ ui-template-pv_set_export_limit_button_style_1.yaml
└ ui-template-pv_get_export_limit_button_style_1.yaml
└ ui-template-pv_set_datetime_button_style_1.yaml
└ ui-template-pv_reset_export_watchdog_button_style_1.yaml

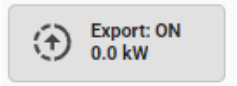
Examples (four control buttons):



Button for ON / OFF export limit

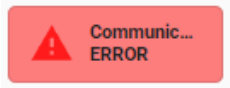
Export limit is OFF

Button is pushed, communication with device in progress, status: waiting



Export limit is ON

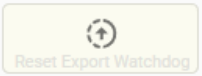
Error, e.g. communication failure



Button Reset Export Watchdog

Button is ready to use

Button is pushed, communication with device in progress, status: waiting



Error, e.g. communication failure

The other buttons such as **Set Export Limit** and **Get Export Limit** are made in the same way.

6. Conclusion

If you get an idea how to improve this integration or if you identify an error, please describe it and the conditions under which it occurred and send it to me by email to address: jozef.moravcik@moravcik.eu

Thanks in advance

Jozef Moravčík

https://www.linkedin.com/in/jozef-moravcik/
https://github.com/Dodo7308/goodwe_sec1000