renaissance superEMS™

user manual

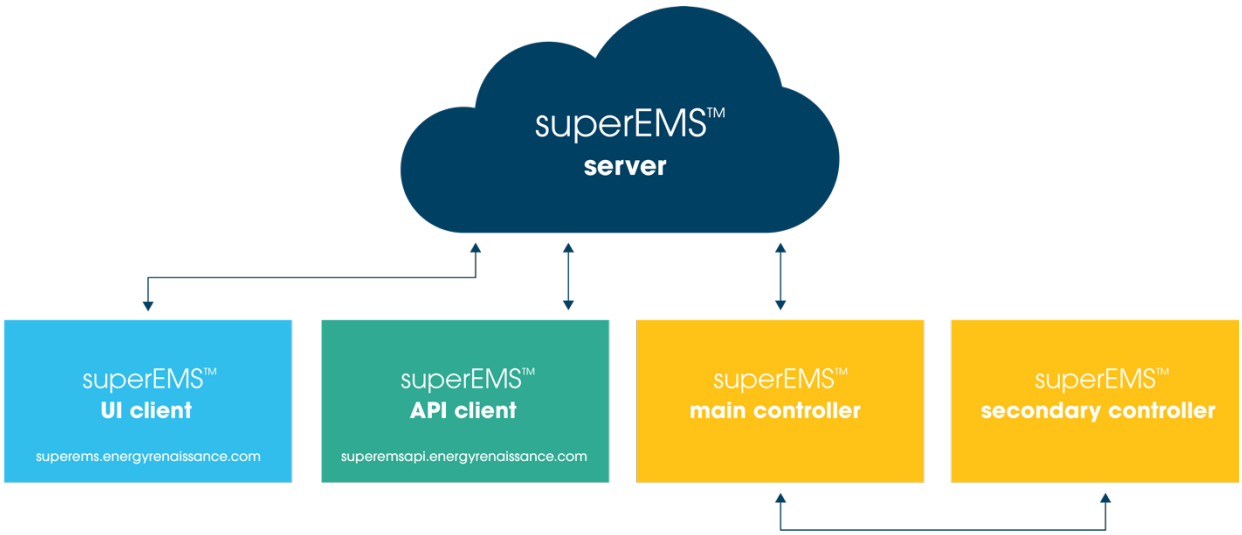
April 2024
review date: 01/07/2024

energy renaissance

energyrenaissance.com

Introduction

The superEMS™ is a superior solution to energy management of battery racks that uses cloud computing to its full to enable remote management both manually via a UI and automatically via an API. The architecture is shown below.



The architecture is a client-server architecture well known from the web which enables the use of a standard web protocol (HTTPS). All the data transfers are encrypted for cyber security.

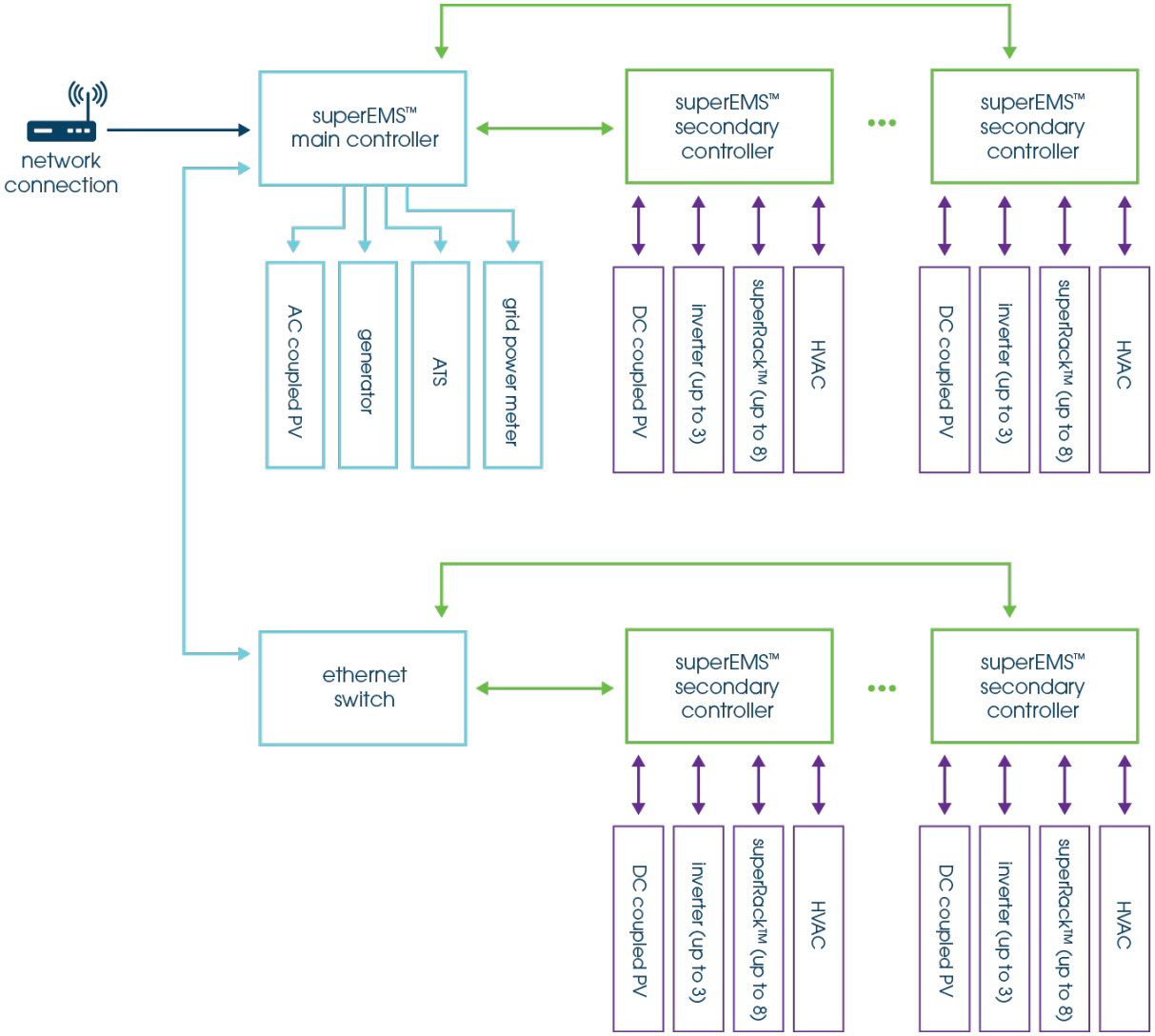
This document describes installation of the hardware (main and secondary), using the UI and using the API ('Appendix D: JSON API Specification and Example Python Code').

superEMS™ main controller vs superEMS™ secondary controller

The superEMS™ main controller and superEMS™ secondary controller are similar but have hardware differences. The secondary controller has an ethernet switch, relay module, processor and USB hub, whereas the main controller has a processor and an ethernet router only. Therefore, the main controller is used to connect to the internet to provide monitoring and control via the client UI and/or JSON API.

As an aside: The difference between an ethernet switch and a router is the security offered, a router can make a sub-net with fixed IP addresses and connect them to the internet through a single connection and the router provides cyber-protection against attempted attacks on the devices connected to its sub-network.

A superEMS™ secondary controller talks to up to 8 superRack™s, 3 battery inverters, 3 DC connected PV converters, and an air-conditioning unit (HVAC).A superEMS™ main controller talks to the internet, a grid power meter, a generator, automatic transfer switch (ATS), an AC connected PV system, and to the superEMS™ secondary controllers:



The superRack™ outdoor comes pre-fitted with a superEMS™ secondary controller. For superRack™, the superEMS™ secondary controller is supplied separately in its own outdoor enclosure. In both cases the superEMS™ main controller is supplied separately in its own outdoor enclosure. Wiring of the superEMS™ main and secondary controllers to the superRack™, inverters, and DC connected PV is covered in the superRack™ installation manual, but for convenience is summarised below.

Communication wiring for superEMS™ secondary controller

Communication wiring for inverter

The communication wiring required by an inverter is typically an ethernet connection back to the superEMS™ secondary controller, though exact cabling is inverter dependant. If multiple inverters are paralleled to increase power, then the inverters require a synchronisation cable between them, the exact cable depends upon the inverter used (but it is often a CAN bus cable).

Communication wiring for superRack™

The superRack™ requires Modbus, USB, and a relay contact to its superEMS™ secondary controller.

Communication wiring for superRack™ outdoor

An outdoor rack is supplied with a superEMS™ secondary controller pre-wired to the rack and inverter (if present). Therefore, only wiring to superEMS™ main controller is required.

For comprehensive wiring instructions please refer to your superRack™ or superRack™ outdoor installation manual.

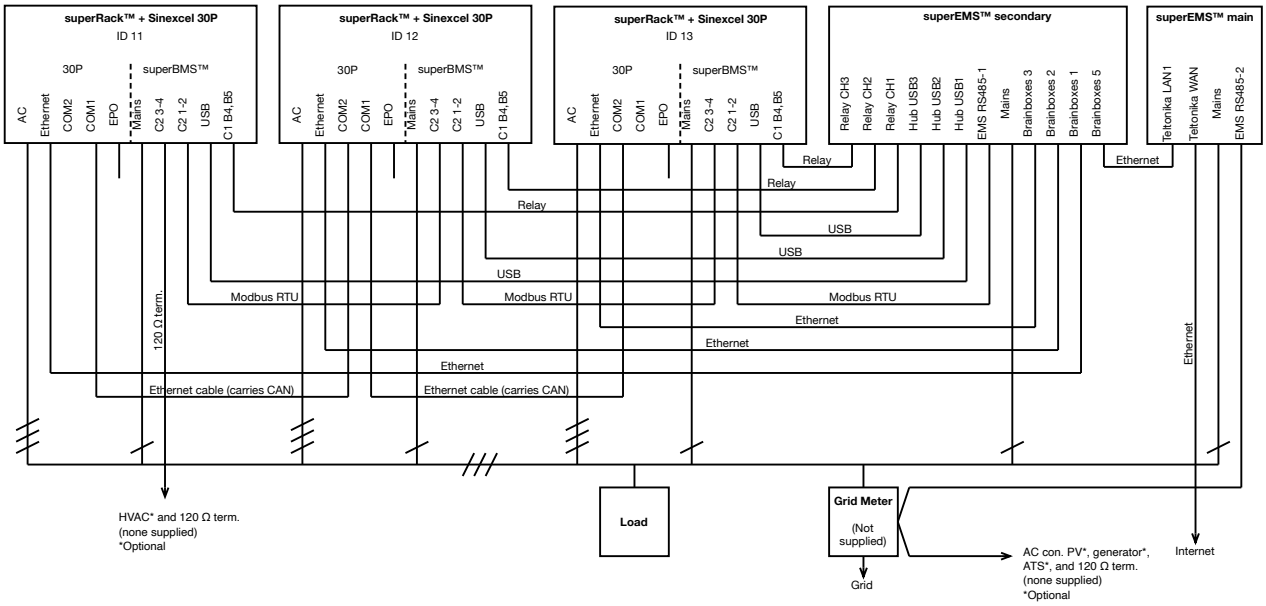
Communication wiring for superEMS™ main controller

superEMS™ secondary controllers are connected to the superEMS™ main controller via ethernet. Similarly, the internet is connected via ethernet. The connections to the grid power meter, generator, Automatic Transfer Switch (ATS), and AC connected PV are typically via a Modbus RTU (RS485) daisy chain, but sometimes are via ethernet (it depends upon exact model of these devices is used). The main controller comes with two RS485 Modbus RTU ports (labelled RS-485-1 and RS485-2) and 2 free ethernet Modbus TCP ports, if this is insufficient then external, not supplied, expanders (hub for RTU expansion and ethernet switch for TCP expansion) will be required. Spare ethernet ports inside superEMS™ secondary controllers may alternatively be used for ethernet expansion instead of or in addition to an external ethernet switch.

For comprehensive wiring instructions please refer to your superRack™ or superRack™ outdoor installation manual.

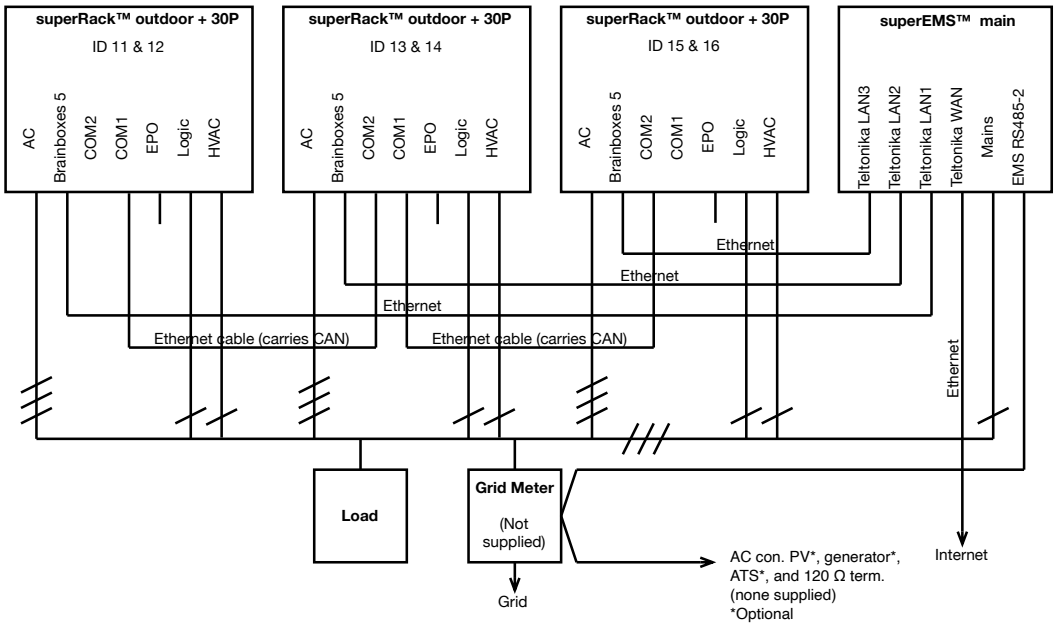
Example wiring superRack™

An example of a system consisting of three superRack™ units, with optional Sinexcel 30P inverters fitted to each of the racks, is given below:



Example wiring superRack™ outdoor

An example of a system consisting of three superRack™ outdoor units, with optional Sinexcel 30P inverters fitted to each of the racks, is given below:



Safety

Ensure you have read and installed your product as per the relevant installation manual which can be found on your client portal before continuing.

	An extremely dangerous power hazard exists during battery energy system installation and connection. Take extreme caution during this process. Failure to do so may cause serious injury or death. Batteries are a constant power supply and should always be deemed to be a live source of energy.
	The battery pack should not be disposed of with household waste at the end of its working life.
	Read the manual before installing and operating the battery pack.
	Keep the battery module away from open flame or ignition sources.
	Wear appropriate personal protective equipment when dealing with the battery pack. Safety boots are required when lifting packs. Insulating gloves, insulating mat, safety goggles and long sleeved/legged non-flamable clothing for electrical connection.
	Keep the battery pack away from children.
	The battery pack may leak corrosive electrolyte.
	The battery pack may explode.
	The battery packs and superRack™ are heavy enough to cause severe injury. Safety boots are required for installation, connection and are required at all times in the work area.

	The battery pack should be disposed of at an environmentally safe recycling facility.
 **EXPLOSIONS**	Do not subject the battery pack to strong impacts. Do not crush or puncture the battery pack. Do not dispose of the battery pack in a fire. Only use insulated tools when dealing with batteries.
 **RISKS OF FIRE**	Do not expose the battery pack to temperatures in excess of 60°C. Do not place the battery pack near a heat source, such as heating systems. Do not expose the battery pack to direct sunlight. Do not allow the battery connectors to touch conductive objects such as wires or moisture or liquids. Do not short circuit battery packs. Ensure vermin, insects or other pests do not inhabit battery rooms or battery enclosures
 **RISKS OF ARCING**	Do not allow battery connectors (pack or rack) to touch conductive objects such as wires or moisture or liquids.
 **RISKS OF ELECTRIC SHOCK**	Do not disassemble the battery pack/rack. Do not touch the battery pack/ rack with wet hands. Do not expose the battery pack/ rack to moisture or liquids. Keep the battery pack/rack away from children and animals.

Risks of damage to the battery pack/rack

- X **Do not** allow the battery pack/rack to come into contact with liquids.
- X **Do not** subject the battery pack/rack to high pressures.
- X **Do not** place any objects on top of the battery pack/rack.
- X **Do not** expose battery pack/rack to high temperatures, high humidity or dust
- X **Do not** subject the battery pack/rack to short circuiting

DANGER! Emergency situations

Leakages	**CAUTION!** Damaged batteries may leak electrolyte or produce flammable gas. **If you suspect a gas leak, take these actions:** • Immediately quarantine the location and do not allow any personnel near the potentially damaged battery. • Contact emergency services / call fire brigade and follow your site procedures. • Contact your provider for further advice and information. **In case of a fire, make sure that an appropriately rated fire extinguisher is nearby.** • The battery pack/rack may catch fire when heated above 150 °C. **If a fire breaks out where the battery pack/rack is installed, take these actions:** • Extinguish the fire potential before the battery pack/rack catches fire or if smoke is present. • If the battery pack/rack has caught fire, **do not try to extinguish the fire**. • Evacuate people immediately and shut off any connected power systems. • Contact emergency services / call fire brigade and follow your site procedures. If the battery pack/rack leaks electrolyte, avoid contact with the leaking liquid or gas. Electrolyte is corrosive and contact may cause skin irritation and chemical burns. If anyone is exposed to the leaked substance, take these actions: **Inhalation:** Evacuate the contaminated area and seek medical attention immediately. **Eye contact:** Rinse eyes with flowing water for 15 minutes and seek medical attention immediately. **Skin contact:** Wash the affected area thoroughly with soap and water for 15 minutes and seek medical attention immediately. **Ingestion:** Induce vomiting and seek medical attention immediately.
Wet batteries	If the battery pack/rack is wet or submerged in water, do not try to access it. Contact your provider for technical assistance.
Damaged batteries	Damaged batteries are dangerous and must be handled with extreme caution. They are not fit for use and may pose a danger to people or property. If the battery pack seems to be damaged, contact your provider for advice. Do not handle.

superRack™ on/off procedures

If you are attempting an initial start of your system, you MUST book in a time for commissioning with Energy Renaissance. You MUST NOT leave the system on without having it commissioned by Energy Renaissance as this risks damaging your batteries and voiding your warranty.

On

1. Ensure the circuit breaker or other switching mechanism such as AC general purpose outlet for the superEMS™ secondary controller and/or superEMS™ main controller is turned off.
2. Ensure logic (230V single phase general) power is turned on to all devices in your installation except for the superEMS™ secondary controller and superEMS™ main controller.
3. Remove the switchgear cover.
4. Turn on the rack isolating switch, if fitted, and logic supply switches above the IEC cable on the rack. This should cause the rack LED to illuminate.

5. Lastly turn on the circuit breaker for the superEMS™ secondary controller and superEMS™ main controller.
6. If the system is off-grid (i.e., a microgrid) then manually start the generator in off-grid mode.
7. To confirm power to the superEMS™s there will be LEDs lit on the superEMS™ processor (labelled superEMS) inside each superEMS enclosure.

Note: If no LEDs are flashing, de-energise and check your installation manual to ensure all cabling connections are correct.

After powering up and initial commissioning is completed, the superEMS™ secondary controller will immediately attempt to start the system.

On rack power up:

1. System check: Rack LED is amber for about 1 s.
2. System waiting for rack-on command from the superEMS™ main controller: Rack LED flashes red.

On rack-on command (after powering up the superEMS™ both main and secondary controllers):

1. Rack LED solid red (for about 10 s, whilst the below contactor sequence completes).
2. Negative contactor closes and check is made that current is zero (earth fault detection).

3. Pre-charge contactor closes for 1 s:

   a) if PCS voltage equals battery voltage close positive contactor and open pre-charge contactor.
   b) else open precharge contactor, wait 100 s and try closing precharge contactor again (repeat step 3).

4. Rack LED solid green to indicate normal operation.

Refit the switchgear cover.

The rack contactors will not close if there is a fault or critical warning (items listed in 'Appendix A: Faults' also applicable if fault develops whilst running):

1. The rack LED goes solid red.
2. Faults and warnings are reported in the user interface at superems.energyrenaissance.com and are described in 'Appendix A: Faults' and 'Appendix B: Warnings'.

See section 'Commissioning' to commission your system.

Off

Your system is equipped with a timeout function, the safest way to stop operation of the system is to turn off the AC circuit breaker or general purpose outlet to the superEMS™ secondary controller.

This effectively times out the inverters so they no longer transfer current and shortly after the battery management system (BMS) will open the battery contactors. Care must be taken to shut off the system as instructed as removing logic power to the racks while there is current flow will damage the internals of the switchgear and can void your warranty.

After 10 seconds it is safe to remove AC power to the rest of your system.

Note:

- that removing power from the superEMS™ secondary controller or superEMS™ main controller for extended periods can void warranty.
- The above instructions are not to be used as a sole isolation, for purposes requiring isolation a risk assessment must be completed and sufficient additional isolations specific to your installation must be completed.
- DC voltage will remain at the terminals of the rack and your PV installation if fitted.
- Turning off the supply to your superEMS™ main controller only will **not** shut down or cause timeouts on your system.

superRack™ outdoor on/off procedures

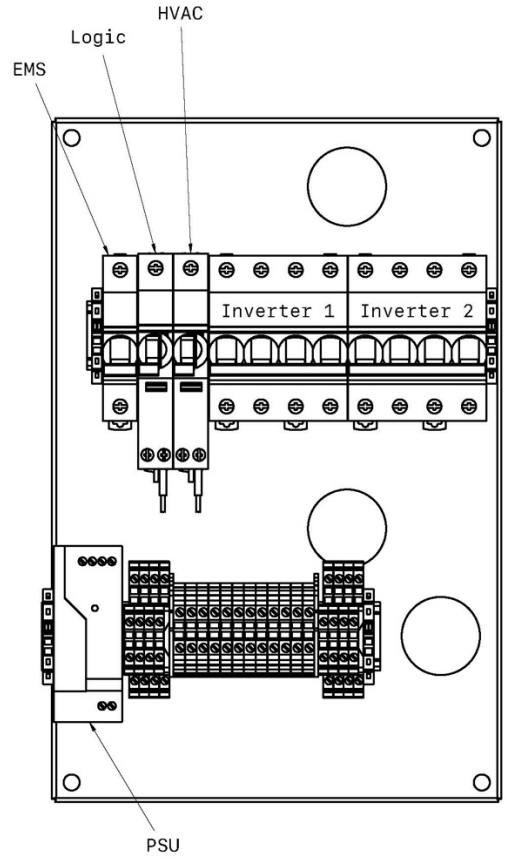
If you are attempting an initial start of your system, you MUST book in a time for commissioning with Energy Renaissance. You MUST NOT leave the system on without having it commissioned by Energy Renaissance as this risks damaging your batteries and voiding your warranty.

On

1. Ensure the superEMS™ secondary controller in each outdoor unit is turned off (see figure below) and ensure that the overall superEMS™ main controller (if fitted) is also turned off (located separately to outdoor units typically in the main switchboard but can also be in a separate communications cabinet).
2. Ensure logic (230V single phase general) power is turned on to all devices, except the superEMS™ secondary controller as detailed in point 1 above, ensure 3-phase power is also on.
3. If the system is off-grid (i.e., a microgrid) then manually start the generator in off-grid mode.
4. Turn on each superEMS™ secondary controller in each superRack™ outdoor.
5. Turn on the superEMS™ main controller.
6. To confirm power, there will be LEDs lit on the superEMS™ secondary controller's processor.
7. If no LEDs are flashing, check your installation manual to ensure all cabling connections are correct.

After powering up and initial commissioning is completed, the superEMS™ secondary controllers will immediately attempt to start the system.

See section 'Commissioning' to commission your system



Off

The system is equipped with a timeout function, the safest way to stop operation of the system is to turn off the 230 V circuit breaker feeding your superEMS™ main and secondary controllers.

This effectively times out the inverters and the superRack™s so they no longer transfer current. Care must be taken to shut off the system as instructed as removing logic power to the racks while there is current flow will damage the internals of the switchgear and can void your warranty.

After 10 seconds it is safe to remove AC power to the rest of your system.

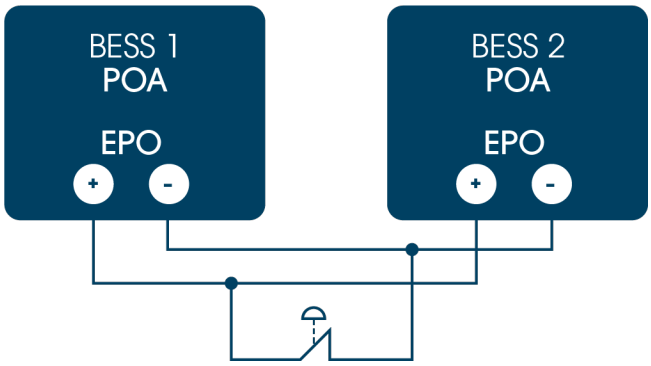
Note:

- that removing logic power from the superEMS™ secondary controller or superEMS™ main controller for extended periods can void warranty.
- The above instructions are not to be used as a sole isolation, for purposes requiring isolation a risk assessment must be completed and sufficient additional isolations specific to your installation must be completed.
- DC voltage will remain at the terminals of the rack and your PV installation if fitted.
- Turning off the supply to your superEMS™ main controller only will **not** shut down or cause timeouts on your system.

superRack™ or superRack™ outdoor emergency shutdown procedure

Inverters have EPO functions for emergency shutdown. The EPO loop should be closed circuit for normal operation and open circuit for a fault. Outdoor units are provided with an Emergency Stop button on their front and the EPO circuit is provided in your point of attachment for you to add additional dry contacts from your own safety/ emergency circuit. Ensure there is no other voltage source introduced on the line.

Ensure your external EPO circuits are healthy before turning on your system as without this your cabinets will not be able to be operated. Checking can be achieved by removing your external wiring to the terminals in the point of attachment and measuring the ohms of your connecting cables. Anything above a short will interrupt operation. Ensure there is no other voltage source introduced on the line.



Commissioning

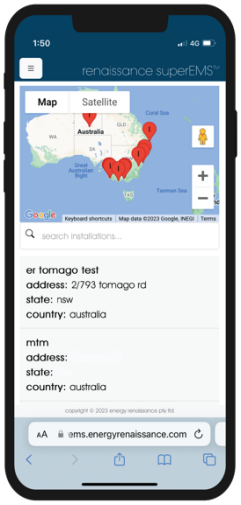
For full explanations of your user interface please see section User Interface Explanation below.

- Ensure all your electrical connections are correct, if not please review your installation manual for correct installation.
- On your web browser go to: https://superems.energyrenaissance.com.
- Enter your email and password that was assigned to you as part of your installation documentation. If you need a login or need to update your password contact Energy Renaissance on 1300 472 020, email service@energyrenaissance.com, or visit energyrenaissance.com/service to fill in a service and support form.
- My profile: contains your personal details including name, email, and mobile number.
- Sites: Your sites will automatically appear in a list once you are logged in. Your battery system will already be set up as a site. Click on your site to see your installation dashboard. You can return to your sites selection at any time by pressing the "sites" tab on the side menu.
- On your dashboard under "notifications" you will see any outstanding faults. If there is only a green tick your system is healthy. If there is a red cross this indicates a fault on your system and will need to be investigated. Click on your "notifications" tab to see all current and previous faults and warnings. Please refer to 'Appendix A: Faults', 'Appendix B: Warnings' and 'Appendix C: Information' for explanations and possible rectifications.
- Settings: Your battery system settings will be pre-populated. If any of these details are incorrect, including your time zone as this is the local time the scheduling will operate by, please contact Energy Renaissance on 1300 472 020, email service@energyrenaissance.com, or visit energyrenaissance.com/service to fill in a service and support form.
- After completing the above steps if your system is still not operational, please contact Energy Renaissance on 1300 472 020, email service@energyrenaissance.com, or visit energyrenaissance.com/service to fill in a service and support form.

User interface (UI) explanation

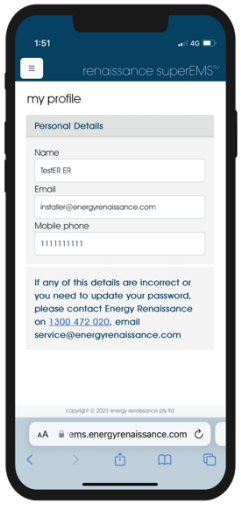
login screen

Open your web browser and enter superems.energyrenaissance.com Here is where you enter your login details or access the demo site (top right button). The 'energy renaissance' title or logo will take you to the main Energy Renaissance website where you can view company and product information. After entering your login information, please click the login button.



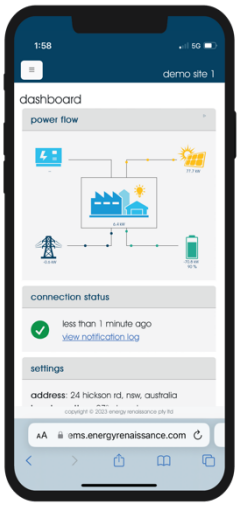
sites

After logging in you will automatically be directed to the 'sites' page. Your sites will be listed, and their locations pinned on the map. To view your specific site information, click on the site you wish to view, and you will be taken to that site's dashboard (see section dashboard below).



my profile

Clicking the 'my profile' tab will take you to user specific information. Ensure these details are correct as your email address and mobile number are where alerts that are time sensitive will be sent you. If you need to change your password or any other information please contact Energy Renaissance on 1300 472 020, email service@energyrenaissance.com, or visit energyrenaissance.com/service to fill in a service and support form.



dashboard

Once you have clicked on your site that you wish to view you will be directed to that site's dashboard. This dashboard gives you an overall insight into how your site is performing.

power flow

In the 'power flow' section it gives you real time information about your site power usage and generation. See below for icon explanations.

generator **solar** **grid** **battery** **load** **The animation indicates the direction of power flow**

The value in kW below each icon indicates the real time power transfer.

Note:

Negative values for the **battery** indicate the battery is charging.

Positive values for the **battery** indicate the battery is discharging.

Negative values for the **grid** indicate there is export to the grid.

Positive values for the **grid** indicate usage from the grid.

The % value below the battery gives the sites State of Charge (SoC) of the lowest battery in that installation.

connection status

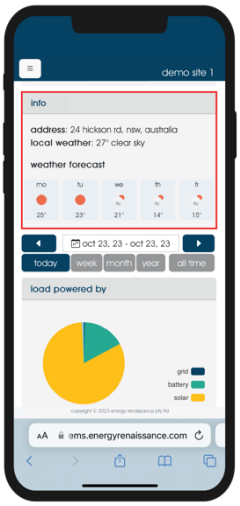
In the ‘connection status’ section you will be able to see if your system is currently healthy, if there are any alerts or faults. To view more information on any current notification either click on the ‘view notification log’ within this section or click on the ‘log’ tab on your side menu.

Indications are as follows:

System healthy

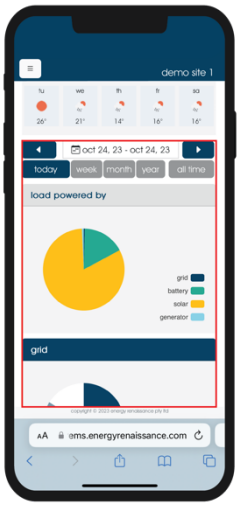
System alarm

System fault

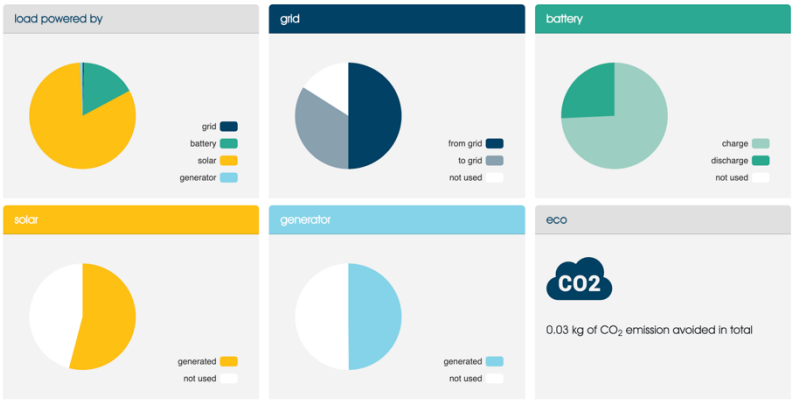


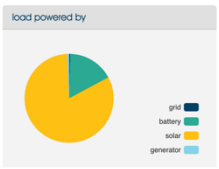
info

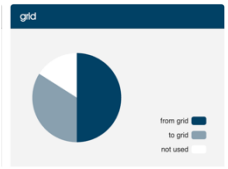
The ‘info’ section gives you the address, local time, and weather forecast information. If any of these details are incorrect, including your local time, please contact Energy Renaissance on 1300 472 020, email service@energyrenaissance.com, or visit energyrenaissance.com/service to fill in a service and support form.

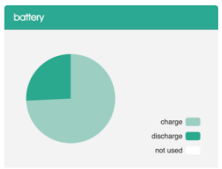


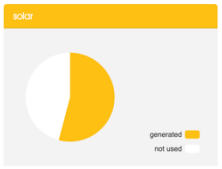
pie charts

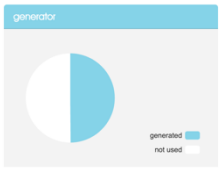


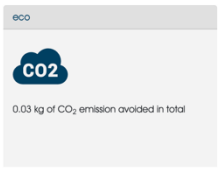












Scrolling down in your dashboard will bring you to your pie charts. These can be filtered by day, week, month, year, or all time (from installation).

The '**load powered by**' pie chart will indicate where your power usage has been generated from. Each category is colour coded, clicking/hovering over the segment or key will indicate the category and display the kWh usage.

The '**grid**' pie chart reflects the information provided by your power meter. Within the selected period it will display the energy that has been used from and exported to the grid over a time scale. Clicking/hovering over the segment or key will indicate the category and display the kWh usage.

The '**battery**' pie chart reflects the information provided by your in-rack battery management system (BMS - superBMS™). Within the selected period it will display the energy that has been used to either charge or discharge your battery over a time scale. Clicking/hovering over the segment or key will indicate the category and display the kWh usage.

The '**solar**' section is time scaled. It will show you how much of the filtered time there was or was not PV (solar) generation. Clicking/hovering over the "time in use" segment or key will indicate the kWh generated for that time-period.

The '**generator'** section will show you how much of the filtered time the generator was in use. Clicking/hovering over the "time in use" segment or key will indicate the kWh generated for that time-period.

The '**eco**' section shows you how much CO2 emissions have been avoided for the filtered time-period; the formula used is 1.06 kg CO_2-e / kWh (1.06 kilograms of carbon dioxide equivalent emissions per kilowatt hour saved). This is the mandated formula in Victoria Australia as of 2022.

Collectively this data can be analysed to detect if there could be improvements to your system. For example, if your 'load powered by' section had a large segment indicating it was being powered by the grid then you could assume that your system could benefit from either more PV or more batteries. If you suspect this is the case, please contact sales@energyrenaissance.com.

energy and SoC graph

The energy graph provides comparative information scaled by the same day, week, month etc information as above in the pie charts.

Your x-axis is your time scale, the LHS y-axis indicates your kWh and the RHS y-axis indicates your battery system's collective State of Charge (SoC).

The legend at the top of the graph allows you to deselect or select based on what information you wish to view. When deselecting the y-axis will automatically scale to make the best use of the space.

The download icon will allow you to download a csv file showing the data record history for the selected time period. See Appendix D for a description of what each column in the file means.

Example 1:

Showing only the battery discharged values.

Example 2:

Hovering over/clicking the bars of the graph will give you more information relating to that time.

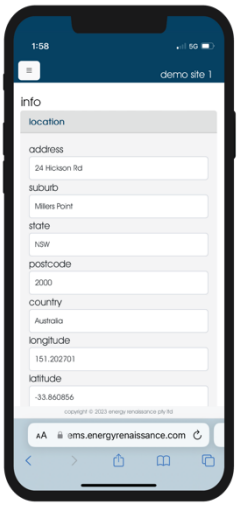
Example 3:

The graphs can be used to see how your system was performing at a moment in time. Here we can see that during the peak hours of PV generation there was no power coming from the grid and the batteries were able to charge to 100% so we can deduce that:

- There is sufficient PV to support the load and charge the batteries.
- There is sufficient battery capacity to support the load during the times there is no PV generation.

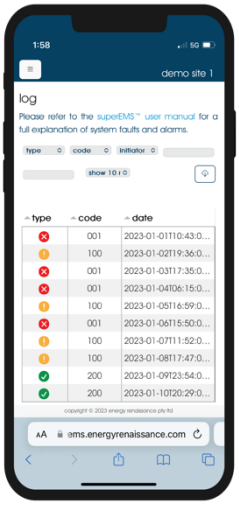
Conversely:

- If the batteries were to continuously remain at a low SoC and there was energy usage from the grid, a fair assumption would be that your system requires more PV to make better use of your batteries.
- If your batteries were to continuously charge relatively quickly during times where there is PV generation and additional energy is exported to the grid, a fair assumption would be that your system requires more battery capacity.



info

The info tab brings you to your site's information. This information will come pre-populated. If any of the information in this tab is incorrect, please contact Energy Renaissance on 1300 472 020, email service@energyrenaissance.com, or visit energyrenaissance.com/service to fill in a service and support form.



log

The log tab will give you details on past and present warnings, faults, and scheduling changes (notifications).

You will see a code, alert, date, and time of the event, resolved date and time if relevant and who the event was initiated by (this will either be a system generated event which will alert you of which device within your system has initiated a notification, or in the case of schedule changes the initiated by will include the username of the person who actioned the change). All of which, excluding time, can be filtered by the drop-down menu at the top. The arrow beside each heading will arrange the history into either ascending or descending order.

Scheduling change
The code identifies either a creation or deletion of a schedule. In the scheduling section you can click on "view schedule history" section to view exactly what changes took place.

System warning
System warnings are for information only and will not inhibit system operation. For more information on system warnings, see 'Appendix B: Warnings'.

System fault
System faults are to notify the user that the system is no longer in operation due to one of the faults listed in Appendix A. If your system is still in fault after you have completed the steps suggested to rectify the fault 'Appendix A: Faults' please contact your installer.

The download icon will allow you to download a csv file showing the log of notifications for the selected time period. See Appendix D for a description of what each column in the file means.

schedule and system operation

The schedule tab is where you can set your system's schedule of events. More complicated schedules and repeated changes to schedules will be easier via the API client ('Appendix D: JSON API Specification and Example Python Code').

To add a new event to the schedule, press the + button and a new event map will appear. Here you can select to either charge or export your system within the selected times you nominate. You can select the days of the week you would like the event to be effective, note, it will be recurring. If you are wanting it to only be effective for a particular day and not recurring, you will have to delete the event after the event's time has passed. You cannot save an event without nominating the day(s) of the week it will repeat on.

After putting in all your information ensure to click the save button as the event will not take effect without doing so.

The events are automatically prioritised in order from top to bottom. Any new event will be created at the end of the schedule (the event priority list is top to bottom). In the case of having 2 or more conflicting events, you can prioritise them by pressing the prioritise up or down arrows.

You do not need a schedule to operate your system, when there is no active schedule, the system will run in automatic mode as detailed in the sections below.

Grid overview

In grid-tied applications the grid will only supply the shortfall of the load, ensure the health of the batteries, charge your batteries when scheduled, export when scheduled, or export any excess PV generation up to your export limit.

Generator overview

If your installation has a generator the generator will only turn on in the case of protecting the batteries in an off-grid applications or in either grid-tied (if coupled with an approved ATS) or off-grid turn on if you have scheduled to charge from the generator.

If your grid-tied system includes a generator it is recommended to include an approved fast action automatic transfer switch (ATS) to ensure your system upholds AC while transferring to and from grid tied, grid forming and generator. Without doing so will require momentary loss of 3 phase power with the use of a suitably rated UPS to uphold single phase logic while the switching takes place, manual starts of your generator and can cause warranty issues if in combination with other conditions.

Battery overview

The batteries will follow instructions based on the scheduling requirements as outlined in the system mode scheduling sections below. The batteries have system overrides for both grid-tied and off-grid systems as outlined below.

Grid-tied systems

If your grid-tied system loses the grid your whole system will shut down until the grid has been restored. During this time, you will not be able to utilise your PV or battery power.

In on grid applications as part of the automatic process to preserve the health of your batteries, regardless of any schedule set, if the lowest cell voltage is 3.0V or less for more than 6hrs the batteries will then charge from the grid or generator up to 30% SoC.

Off-grid systems

In off-grid applications that include a generator as part of the automatic process to preserve the health of your batteries, regardless of any schedule set, if any cell in any rack reaches 3.0V (approx 10% SoC) or below the system will turn on the generator and charge all the racks to at least 30% SoC. This will repeat itself in every instance. You must ensure that your generator is without faults, able to be operated at all times and that you have correctly sized your system so that the generator cannot supply the maximum load the generator may trip and then the load will have to be manually disconnected and the generator manually restarted (black start). (see 'Appendix F: Black or initial start of your system (off-grid)').

When the generator is operating, any AC connected PV is turned off so that the combined power of the PV and generator does not exceed the battery inverter power.

Note *that leaving your system off for extended periods can cause warranty issues.*

System “automatic” mode

If there are no active events in the schedule the system will run in automatic mode, which is explained below.

Battery operation in automatic mode

The battery operation in automatic mode will be to charge or discharge the batteries depending on the load requirements or available generation power.

Generator operation in automatic mode

The generator in automatic mode will remain off.

Grid operation in grid-tied-automatic mode

The grid in automatic mode will export excess PV to your export limit and import any shortfalls required of the load.

System “charge” mode

You have the option to charge from either the grid or generator (if applicable). Scheduling a charging event can be useful to ensure that the system overrides to protect the batteries do not take place at inconvenient times such as, turning on the generator in the middle of the night or charging from the grid during peak tariff times.

When scheduling a charging event, you must choose where you would like to charge from and the days and times, you’re wanting to charge your system. **You will not be able to save your new or modified event if no options are selected.**

Battery operation in charge mode

Charging from either the grid or generator will charge the batteries at system maximum capacity at that point in time. The batteries will not discharge to support the load again until the scheduling time has lapsed.

Generator operation in charge mode

When a system charge event is scheduled to charge from the generator the generator will turn on at the nominated time and will support the load requirements (along with PV if fitted) and charge the batteries up to maximum system capacity and remain on until the scheduled time has lapsed. If off-grid, any AC connected PV fitted will be turned off before the generator starts to prevent the combined power of the generator and PV exceeding the capability of the battery inverter.

Grid operation in charge mode

When a system charge event is scheduled to charge from the grid the grid will support the load requirements from the nominated time (in conjunction with PV if fitted) and charge the batteries up to maximum system capacity and remain to support the load until the scheduled time has lapsed.

System "export" mode (grid tied applications only)

Battery operation in export mode

When selecting export, the batteries will discharge to support the load and excess will be exported to the grid at maximum system capacity or your export limit, whichever is less. The batteries will not charge with excess PV again until the scheduled time has lapsed unless the PV can supply the load and the export limit as well as charging the batteries.

Generator operation in export mode

The generator will remain off while system is in export mode.

Grid operation in export mode

The grid will accept export of all PV generation and battery storage up to maximum system capacity or your export limit, whichever is less. The grid will continue to export excess PV generation and support the load even after the batteries have been fully discharged until the scheduled time has lapsed.

The download icon will allow you to download a csv file showing the schedule of events for the selected time period. See Appendix D for a description of what each column in the file means.

Examples of schedules

Name	On/off grid?	Description	Setting
PV sponge	On	Soak up all the excess PV instead of using the grid	Nothing – normal operation
Ensure battery charged at end of day	On or off	The batteries are required to be charged because their energy is required overnight.	1. Look up sunset time, T_s (to nearest hour rounded down in 24 h format). 2. Lowest-average, PV power, P_s in kW (could be 0). 3. Maximum-allowable, daytime, import power (grid or generator), P_g in Kw. 4. Highest-average load power, P_l in kW. 5. Worst-case, battery energy required overnight, E_b in kWh. 6. Schedule charge from $T_s - E_b$ / ($P_s + P_g - P_l$) to T_s.

Appendix A: Faults

Note: Not all faults will be applicable to every installation. Only installations with approved devices will have the compatible communications with the superEMS™ to flag faults.

Note: Faults will inhibit operation. Investigate and rectify all faults before attempting operation.

Code	Fault	Description	Possible causes	What to do
01	Network	The superEMS™ has been disconnected from the network	• Provider outage • Modem failure	• Check with your network provider to see if the network is down • Restart your modem
02	Generator	The generator has not started after superEMS™ instruction	• The generator has ran out of fuel • The generator has some other error	• Inspect the fuel gauge of the generator • Check the generator for any other faults
03	Inverter fault status	The inverter has an active fault	• Any fault of the inverter will flag this error • This fault will be flagged if the inverter is in a state not accepted by the superEMS™	• Fault find by identifying the fault from possible alerts listed below • Ensure the inverters are in remote mode if applicable
04	DRM0	DRM0 function of the inverter has been activated	• Estop has been activated • External DRM0 circuit has been activated	• Deactivate the e-stop • Remove your external DRM0 connections and test for continuity
05	EMS comms timeout	Device has lost communications with the superEMS™	• Devices have been turned off • The system has lost 230V AC power	• Ensure the system has its 230V AC and 12V DC circuit breakers turned on • Ensure there is supply voltage to all your devices
06	AC Phase reversed	reversed to the inverter	• Phase reversed	• Isolate and disconnect your point of attachment and switch 2 of the 3 phases

Code	Fault	Description	Possible causes	What to do
07	AC Under frequency	AC under frequency to the inverter	• Phased reversed • Frequency of the generator is set too low	• Isolate and disconnect your point of attachment and switch 2 of the 3 phases. • Set the frequency of the generator to 50Hz
08	AC Over frequency	AC over frequency to the inverter	• Frequency of the generator is set too high	• Set the frequency of the generator to 50Hz
09	AC Under voltage	AC under voltage to the inverter	• Voltage of the generator is set too low	• Set the generator voltage to 3 phase 400V
10	AC Over voltage	AC Over voltage to the inverter	• Voltage of the generator is set too high	• Set the generator voltage to 3 phase 400V
11	Battery mismatch	The battery is not seeing the required DC voltage for the number of packs allocated to the installation	• Loose connection on one or more of the power connectors on the rack	• Consult with an electrician to check to rack power connections
12	Battery isolator	The battery isolator is engaged	• The isolator is in the off position	• Consult with an electrician to turn on the in rack isolating switch
13	Pack temperature	One of the cells has reached the upper temperature limit	• There is insufficient cooling in the battery room or outdoor unit	• Ensure your HVAC is correctly rated, operational and sufficiently cooling
14	Cell Vol low	One of the cells in the superRack™ has hit its critical low voltage	• The system has been over discharged • The system has been left at a low SoC for too long	• Contact your installer
15	Cell Vol high	One of the cells in the superRack™ has	• The system has been over charged	• Contact your installer

Code	Fault	Description	Possible causes	What to do
		hit its critical high voltage		
16	Cell vol differential	There is too great a spread of voltages between your cell min and max voltages	• Deterioration of a cell • The system has previous been stressed to its limits	• Contact your installer
17	Cell temp differential	There is too great a spread of temperatures between your cell min and max temperature readings	• The in-rack fans aren't operating properly • There is insufficient cooling in the battery room or outdoor unit	• Ensure your HVAC is correctly rated, operational and sufficiently cooling • Contact your installer
18	HVAC high/low temp	The HVAC system has a high or low temp fault	• The HVAC system has overheated or cooled	• Contact your installer
19	HVAC sensor	The HVAC system has a sensor fault	• There is a failure of a sensor	• Contact your installer
20	HVAC humidity	The HVAC system has a high or low humidity fault	• The environmental humidity levels are too high or low	• Attempt to operate system once humidity levels have improved • Contact your installer
21	Compressor failure	The HVAC system has a compressor failure alarm	• Component failure	• Contact your installer
22	HVAC fan failure	The HVAC system has an internal or external fan failure alarm	• Component failure	• Contact your installer
23	EPO	The inverter has registered an EPO alarm	• The E-stop of your outdoor unit could be activated. • Your external e-stop if fitted	• Investigate all e-stops • Investigate your external EPO circuit.

Code	Fault	Description	Possible causes	What to do
			could be activated	

Appendix B: Warnings

Note: Not all warnings will be applicable to every installation. Only installations with approved devices will have the compatible communications with the superEMS™ to flag warnings.

Note: Warnings will not inhibit operation but will likely derate your unit while in operation until Warnings have been cleared. Some warnings are part of standard operation i.e., cell voltage warnings and require no action, other warnings if frequent could be cause for investigation i.e., operating temperatures.

Code	Warning	Description
100	Rack vol high	A rack has reached its high voltage warning threshold
101	Rack vol low	A rack has reached its low voltage warning threshold
102	Cell vol high	One or more of the cells in your battery system has reached its higher cell voltage warning threshold
103	Cell vol low	One or more of the cells in your battery system has reached it lower cell voltage warning threshold
104	Discharge Over Current	the discharging current of your battery has reached its over current warning threshold
105	Charge Over Current	The charging current of your battery has reached its over current warning threshold
106	Rack temp high	The rack temperature has reached its high warning threshold
107	Rack temp low	The rack temperature has reached its low warning threshold
108	Batt SoC high	The battery SoC has reached its high warning threshold
109	Batt SoC low	The battery SoC has reached its low warning threshold
110	Cell vol differential	The spread of voltage between your cell min and max voltages have reached its warning threshold
111	Cell temp differential	The spread of temperature between your cell min and max temperature readings has reached its warning threshold
112	Inverter warning status	Your inverter is in warning status
113	Inverter derating status	Inverter cannot run at full capacity

Appendix C: Information

Note: not all informational notifications will be applicable to every installation. Only installations with approved devices will have the compatible communications with the superEMS™ to flag information.

Code	Info	Description
200	Schedule change	The site's schedule of events was changed.

Appendix D: JSON API Specification and Example Python Code

The JSON API client allows control of the superEMS™ from a computer rather than direct control (which the UI client allows). The API client has feature parity with the UI client.

There is a Python module that is a simpler alternative to the raw JSON API; both JSON and Python forms are described below. Since the Python form is simpler to understand it is given 1st and therefore acts as an introduction to the JSON API. If intending to use the JSON API it is recommended that you read both descriptions, Python and then JSON. Authorization is always required and in addition the description of the authorization below provides useful background to the dataclasses used for all the methods, therefore it is recommended to read this description 1st.

To use either API an ER supplied ID and Secret are required (same ID and Secret for both APIs), they are obtained from service@energyrenaissance.com or in Australia by calling 1300 472 020.

A key feature of both APIs is that time given to methods e.g., a start and end time, has minute precision at most and no time-zone information including no folds (used to distinguish repeated times due to daylight savings). The times are interpreted as local time and if due to added daylight savings a time repeats the results for both times are included. Time data is given if the record/notification/etc.'s time is in the range start <= time < end (i.e., a half open range).

The python module can be downloaded here. It is a robust, yet simple means of controlling a superEMS™. The Python API also handles many common communication issues, like temporary loss of connection, automatically. The module is modest in size, excluding comments less than 1,000 lines, and therefore the easiest way of using the module is to simply copy `superems.py` into your own project. The module also requires the requests library (`pip install requests`). Correct installation can be verified by `python superEMS.py` which executes example code similar to the code described below.

The superEMS™ JSON API client allows users to retrieve data from and setup charge/discharge event schedules for their installations. The base address of the API is https://superemsapi.energyrenaissance.com/v1 and the API follows the usual RESTful conventions utilising JSON for message payloads (bodies) and replies. The API is programming language agnostic and therefore useful if a language other than Python is to be used.

Methods

Authentication

The token returned by the authentication is only valid for 24 hours. The Python API will automatically renew the token, however if the JSON API is used the token requires manual renewal.

Python

To use the Python API there are five requisites:

1. What to do on a communication or authentication error, in particular the number of retries before terminating needs to be decided (default infinite retries). When developing code zero retries is recommended.
2. How to log faults that are to be retried (default print). In many case the default action of printing errors and then retrying will be sufficient. Alternatively, a log function can be specified.
3. The code to execute once authentication is complete. This is the code that carries out the required actions.
4. The ER supplied Secret.
5. The ER supplied ID.

The Python static method inside class superEMS™ is the entry point for the Python API:

```
@staticmethod
def controller(
    *,
    er_supplied_id: str,
    er_supplied_secret: str,
    code: Callable[[SuperEMS], None],
    log: Callable[[str], None] = print,
    num_retries: int = -1,
):
    """
    Log into the superEMS™ server and execute given `code` (passing it a
new superEMS™ client),
    if an `HTTPErrors` occurs `log` the error, wait 60 s, and repeat
process (login then `code` with new client)
    at most `num_retries` times.

    A negative (default -1) `num_retries` means infinite retries on
`HTTPErrors` and 0 retries means do not retry
    on `HTTPErrors` (loop through login and `code` exactly once).

    `er_supplied_id` and `er_supplied_secret` are issued separately for
each user that wishes to write client code,
    contact service@energyrenaissance.com or in Australia call 1300 472
020.
    """
```

A typical call is:

```
sales_demo_id = "4"
sales_demo_secret = "p7EogtHOiCsumjBDvpk8t78csAhIgpnvsFMeiu6M"
SuperEMS.controller(
    er_supplied_id=sales_demo_id,
    er_supplied_secret=sales_demo_secret,
    code=example,  # The code to execute.
    log=print,  # In real code, probably a log function.
    num_retries=0,  # This depends upon the application and if `code` can
be restarted.
)
```

The signature of the function to be executed by the controller is:

```
def example(super_ems: SuperEMS) -> None:
    """Exercise all the superEMS™ methods as an example of what the API can
do."""
```

It receives a superEMS™ object called super_ems from the above call to controller. This object is used inside the function e.g., to access the authentication granted:

```
super_ems.authorization
```

You don't normally need to access the authentication, but it can be useful when debugging. Particularly the time validity of the authentication.

Like all the data available in the Python API, Authorization (returned by the above call) is a dataclass that inherits from JSONPersistable. JSONPersistable has two functions: it identifies the version of the data that is persisted (stored externally) and it provides a conversion to a format suitable for conversion to JSON. JSONPersistable is:

```
@dataclass(order=True, frozen=True, kw_only=True)
class JSONPersistable(ValidateTypes):
    """
    Mixin required methods for persisting using JSON (all the dataclasses
use this as their base class).
    """

    version: str = __version__
    """The API version used to create the dataclass, useful if instance is
persisted and API version changes."""

    def to_json_compatible(self) -> dict[str, Any]:
        """
        Convert `self`, via deep copying, to a `dict` suitable for
`json.dumps`; must be reverse of constructor.
        Useful for persisting or exporting a `dataclass` as JSON.
        Used to write `Schedules` back to the superEMS™ server.

        Default implementation is `return asdict(self)`.
        """
```

Authorization inherits from JSONPersistable and adds the arguments that method authorization was called with and the returned values as fields:

```
@dataclass(order=True, frozen=True, kw_only=True)
class Authorization(JSONPersistable):
    """Login authorization."""

    er_supplied_id: str
    """Identifier for each user that wishes to write client code."""
```

```
    er_supplied_secret: str
    """Secret that corresponds to `er_supplied_id`."""
    token_type: str
    """The type of the token returned, e.g. Bearer."""
    expires_in: int
    """The lifetime in seconds of the access token."""
    access_token: str
    """The token to be given in the header for subsequent interaction with
the API."""
```

Like Authorization the returned dataclasses from the other methods in the API (described below), inherit from JSONPersistable (hence have version strings and are persistable via JSON) and they contain the generating method's arguments as fields.

To create a JSON string, j, from an instance, o, of any of the dataclasses:

```
jc = o.to_json_compatible()
j = dumps(jc)
```

Where dumps is the standard python object to JSON converter function from package json. To convert back from a JSON string, j, to an instance, o, of a JSONPersistable dataclass (all the dataclasses in the module) of type, t:

```
jc = loads(j)
o = t(**jc)
```

Where loads is the standard python JSON to object converter function from package json. All the constructors of the dataclasses, t, accept a spread, **, of a JSON object from loads as their constructor argument.

Raw JSON

POST	/client/auth
It's an Authentication/Authorization method. It generates an API access token that enables the client app to consume the available endpoints.	
PARAMETERS (body)	
NAME	DESCRIPTION
{body} Required ***JSON*** Request's body	JSON structure with the required credentials data: `{ "grant_type": "client_credentials" "client_id": "{{your_client_id}}", "client_secret": "{{your_client_secret}}" }`
grant_type Required ***string*** JSON structure	Must be "client_credentials".
client_id Required ***integer*** JSON structure	ID provided by Energy Renaissance (`er_supplied_id`).
client_secret Required	Secret provided by Energy Renaissance (`er_supplied_secret`).

string JSON structure	
RESPONSE	


```
{
    "token_type": "Bearer",
    "expires_in": 86400,
    "access_token": "eyJOeXAiOiJKV1QiLCJhbGci0iJSUzI1NaJ8.eyJhdWQiOiI0IiwianRpIjoiNWE1Mjc
4MGMyMzMzOWEyMzdiNDcxYTA1ZTIxMzJhMTVhMDA2MTQ0NTk4NTczOWY4Z-*jOGY0YzNiOTFiNjZmZjM0ODgxZTk5
MWE4YWM2Zjk1LCJpYXQiOjE2ODQxMTU4OSEuMTEwMjc2LCJuYmYiOjE2ODQxMTU4ODEuMTEwMjgxLCJleHAiOjE2O
DQyMD-yODEuMDkzMDAxLCJzdWIiOiIiLCJzY25wZXMiOltdfQ.fW3q7krSAaadIolDGG9Syfy2kqIWYdS4EB—aKT
dqmW-rM9K6hkoUgX-DV3M5f8mzwam2H0EgoLBSjgRlE1zGmnKjgsWWkZuRJfrh0nS-DibZDukpHav4dnm75sHs6sI
oZWYFYt_OuZ_eOVPW81F8KL62Kr0Mpt1DZaRPjYdJyAUpT6wetSWqEumKtfGjD_M1jxbRb0MiethT5oGNl9K2_AT2
-yjj1nbdew4SbYZGt9GJ311aAHZsZXDq98ESNsS6btf4IpwV-hRAlivadYsSywsOr0XMNh1NYpaGChBzTzqPLQMvD
SZTrzo0Gw-hvApn5HfklAS9PKrSLSjV9N7NF1Dy334s1Zsa1_XvVhVrhYr3VN2W1RV0nNVPTXNsSEcB9uZTzwuKgh
b6Oi6QBUSSPj50vmGZaN9a5rnkP9VCoj0a93CTdnohqwfMphCcEkaddldfHLpH_69zb3YfXf9fh6iWTWLVcWqC5U0
RLVNhILobjClFUl3qsYjnO8CAq3_5EE_GK2Wq6_9HuKwGeOTGKcpzlzJ4ceeJ1NlxEtTRwH5DF2AQwCUqC55vYwPY
hBk_0m2kR2zPmbxP3gW6WJG3o13nMW_OPfui4opLDBA_wHcM_8zUCXp48TUjrFi02Ck8khtdCR9P27VBar8Uc4ITq
Y-_iYIBrBOBzm_8-3BQ"
}
```


CODE	DESCRIPTION
200	Success
404	Not found. You may need to review the URI you are using on your request.
409	Client Error. You may need to review your request parameters.
500	erver Error. Please notify SuperEMS API team about this type of messages.

List of Sites for a Given Login

Python

For a given ID and Secret there are associated sites. The call:

```
super_ems.sites()
```

Returns a list of Sites and the Site dataclass is:

```
@dataclass(order=True, frozen=True, kw_only=True)
class Site(JSONPersistable):
    """Site; as described in the Project Specification Form."""

    id: int
    """Unique identifier for each site."""
    name: str
    """Name (official site name)."""
    system_location: SystemLocation
    """The street address of the site."""
    system_settings: SystemSettings
    """The main settings used by the superEMS™ (important that these are
correct!)."""
    site_information: SiteInformation
    """General information about the project build."""
    battery: Battery
    """Information about the battery system and the battery inverters
installed."""
    pv: PV
    """Information about the PV (a.k.a. solar) install (if any)."""
    grid: Grid
    """Information about the grid connection (if any)."""
    generator: Generator
    """Information about the generator (if any)."""
    communication_control: CommunicationControl
    """Information about the on-site communications."""
    single_line_diagram: SingleLineDiagram
    """Links to site diagrams including the main site single-line-diagram
(SLD)."""
```

There is considerable information about a site available, see the Python source code for details. The most important items are the site's ID, name, location, and settings. The site ID, referred to as site_id, is used in the other methods to select the site of interest.

Raw JSON

GET	/sites
Retrieves the lists of sites associated to the current user.	
PARAMETERS	
NAME	DESCRIPTION
N/A	Not parameters required.
HEADERS	
Key	Value
Authorization	"Bearer " + access_token

RESPONSE


```
[
    {
        "id": 173,
        "name": "Demo Site 1",
        "system_location": {
            "address": "24 Hickson Rd",
            "suburb": "Millers Point",
            "state": "NSW",
            "postcode": 2000,
            "country": "Australia",
            "longitude": "151.202701",
            "latitude": "-33.860856"
        },
        "system_settings": {
            "number_of_battery_racks": 2,
            "total_battery_size": 120,
            "number_of_battery_inverters": 2,
            "total_battery_inverters_size": 100,
            "total_PV_size": 100,
            "number_of_generators": null,
            "total_generator_size": null,
            "on_grid": 1,
            "is_demo": 1
        },
        "site_information": {
            "project_order_name": null,
            "primary_organisation": null,
            "contract_scope": null,
            "end_customer": null,
            "installer_electrical_contractor": null,
            "engineering_build_contractor": null,
            "communication_control_configuration_owner": null,
            "commissioning_lead": null,
            "project_manager_project_engineer": null
        },
        "battery": {
            "product": null,
            "size": null,
            "location": null,
            "discharge_duration": null,
            "inverter_location": null,
            "inverter_connection_point": null,
            "inverter": null,
            "inverter_quantity": null,
            "inverter_voltage_range": null,
            "inverter_communication_interface": null,
            "inverter_datasheet": null
        },
        "pv": {
            "panel": null,
            "installer": null,
            "size": null,
            "inverter_converter": null,
            "coupling_system": null,
            "inverter_communication_interface": null,
            "inverter_datasheet": null
        },
        "grid": {
            "connection": null,
            "meter": null,
            "meter_current_transformers": null,
```

```
            "meter_communication_interface": null,
            "meter_datasheet": null,
            "is_there_a_grid_export_limit": null,
            "grid_protection_device": null,
            "grid_protection_device_communication_interface": null,
            "grid_protection_device_datasheet": null
        },
        "generator": {
            "generator": null,
            "size": null,
            "controller": null,
            "controller_communication_interface": null,
            "controller_datasheet": null,
            "additional_on_off_contactor": null,
            "transfer_switch": null,
            "transfer_switch_communication_interface": null,
            "transfer_switch_datasheet": null
        },
        "communication_control": {
            "site_communication_system": null,
            "site_energy_controller": null,
            "scope_of_er_comms": null,
            "available_ips": null,
            "number_of_superemss": null
        },
        "single_line_diagram": {
            "site_sld": null,
            "site_layout_and_other_diagrams": null
        }
    },
    …
]
```


CODE	DESCRIPTION
200	Success
404	Not found. You may need to review the URI you are using on your request.
409	Client Error. You may need to review your request parameters.
500	Server Error. Please notify SuperEMS API team about this type of messages.

History of Power Flow and SoC for a Given Site

Python

The power flow on a site and the battery SoC can be obtained for a period by calling the history method, an example call is below (in example, what happened in the last two minutes - note data updated once per minute therefore last minute might be empty).

```
# Timezone unaware, interpreted as the local time at the location where the superEMS™ is sited,
# no seconds, no microseconds, and no folds.
now = datetime.now().replace(second=0, microsecond=0, fold=0)
history = super_ems.history(
    site_id=site_id,
    start=now - timedelta(minutes=2),
    end=now,
)
```

The returned dataclass is:

```
@dataclass(frozen=True, order=True, kw_only=True)
class History(SiteAndPeriod):
    """History of `Record`s associated with a site over a period of time."""

    grid: list[PowerRecord]
    """
    Grid connection quantity, if any, from power meter.

    Constructor also accepts a JSON list as a string.
    """
    battery: list[PowerRecord]
    """
    Battery connection quantity, which are taken from the inverter if there is one otherwise from the superBMS™.

    Constructor also accepts a list converted from JSON.
    """
    pv: list[PowerRecord]
    """
    PV (solar) connection quantity, if any,
    which are taken from the master inverter if there is one otherwise from a power meter.

    Constructor also accepts a list converted from JSON.
    """
    generator: list[PowerRecord]
    """
    Generator connection quantity,
    which are taken from the generator controller if there is one otherwise from a power meter.

    Constructor also accepts a list converted from JSON.
    """
    soc: list[SoCRecord]
    """
    Battery SoC records in percent (regardless of quantity specified!).

    Constructor also accepts a list converted from JSON.
    """
```

The base dataclass, which contains the called method's arguments, is

```
@dataclass(frozen=True, order=True, kw_only=True)
class SiteAndPeriod(JSONPersistable):
    """Site and time period associated with records, logs, and
schedules."""

    site_id: int
    """Unique identifier of the site."""
    start: datetime
    """Records/logs/events' start time, record/notification/event's
datetime >= start datetime."""
    end: datetime
    """Records/logs/events' end time, record/notification/event's datetime
< end datetime."""
```

The power records are:

```
@dataclass(order=True, frozen=True, kw_only=True)
class PowerRecord(TimeStamp):
    """An individual power record."""

    P: float
    """Power in kW (**total** of phases if 3-phase)."""
    PF: float
    """
    Power factor (**average** of phases if 3-phase).

    PF ≅ 1,000 * P / (U * I)

    The above formula for PF would be equal, not approximately equal,
    and hence this measurement would be redundant,
    if single phase or if all the phases in a 3-phase system were balanced,
    i.e. all phases have both the same voltage and current.
    """
    U: float
    """Line to neutral RMS voltage in Vrms (**average** over phases if 3-
phase)."""
    I: float
    """Phase RMS current in Arms (**total** over phases if 3-phase)."""
```

And the SoC records are:

```
@dataclass(order=True, frozen=True, kw_only=True)
class SoCRecord(TimeStamp):
    """An individual SoC record."""

    value: float
    """Battery SoC as a ratio."""
```

Both are which are time stamped:

```
@dataclass(order=True, frozen=True, kw_only=True)
class TimeStamp(JSONPersistable):
    """A time stamp for a record (base class for other Record types)."""

    datetime: datetime
    """
    The local datetime the quantities were recorded at.
    """
```

Raw JSON

GET	/fetch/{site_id}/history
Retrieves all the power and SoC record history associated with the given site applying the requested date filters.	
PARAMETERS	
NAME	DESCRIPTION
site_id Required ***integer*** path	Unique site number from which records are requested.
start Required ***string*** query	Start datetime. ISO 8601 format, no seconds, nor smaller, and no time zone: "YYYY-mm-ddTHH:MM". Returned records start_no_fold <= record_datetime < end_no_fold or start_with_fold <= record_datetime < end_with_fold.
end Required ***string*** query	End datetime. ISO 8601 format, no seconds, nor smaller, and no time zone: "YYYY-mm-ddTHH:MM". Returned records start_no_fold <= record_datetime < end_no_fold or start_with_fold <= record_datetime < end_with_fold.

HEADERS	
Key	Value
Authorization	“Bearer ” + access_token

RESPONSE


```
{
    "battery": [
        {
            "datetime": {
                "day": 1,
                "min": 0,
                "fold": false,
                "hour": 0,
                "year": 2023,
                "month": 2
            },
            "P": 23.23,
            "PF": 0,
            "U": 43.78,
            "I": 98.133
        },
        …
    ],
    "soc": [ … ],
    "generator": [ … ],
    "grid": [ … ],
    "pv": [ … ]
}
```


CODE	DESCRIPTION
200	Success
404	Not found. You may need to review the URI you are using on your request.
409	Client Error. You may need to review your request parameters.
500	Server Error. Please notify SuperEMS API team about this type of messages.

Get Schedule of Events for a Given Site

Python

The events scheduled for a site is obtained from:

```
super_ems.schedule(site_id=site_id)
```

The returned Schedule contains a list of Events:

```
@dataclass(frozen=True, order=True, kw_only=True)
class Schedule(JSONPersistable):
    """List of `Event`s for a site."""

    site_id: int
    """Unique identifier for the site."""
    events: list[Event]
    """
    The scheduled events for the site.
    """
```

And Event is:

```
@dataclass(frozen=True, order=True, kw_only=True)
class Event(JSONPersistable):
    """An event in the schedule (an `Action` with start and end times, a
repeat schedule, and an additional source)`."""

    start: time
    """
    Start time, active when current time >= `start`.
    """
    end: time
    """
    Stop time, active when current time < `stop` to nearest minute
    (stops when current = `stop - 1 minute`).
    """
    action: Action
    """The action to take."""
    repeat: Days
    """Days of the week that the action repeats on."""
    charge_from: AdditionalSource
    """Source of additional power, in addition to PV, to charge the battery
from."""
```

More details in source code.

Raw JSON

GET	/fetch/{site_id}/schedule
It retrieves the schedule for the given installation.	
PARAMETERS	
NAME	DESCRIPTION
site_id Required ***integer*** path	Unique site number from which its schedule is requested.
HEADERS	
Key	Value
Authorization	"Bearer " + access_token
RESPONSE	

```
[
    {
        "start": "09:30",
        "end": "09:00",
        "action": "charge",
        "repeat": 64
    }
]
```

CODE	DESCRIPTION
200	Success
404	Not found. You may need to review the URI you are using on your request.
409	Client Error. You may need to review your request parameters.
500	Server Error. Please notify SuperEMS API team about this type of messages.

Set Schedule of Events for a Given Site

Python

The existing schedule can be overwritten by a new schedule. Typically, the old schedule is read, then modified, and then written, e.g.:

```
old_events = schedule_before_set.events
event = Event(
    # Not using `action` as the actual action;
    # instead toggling it for tracking purposes between before and after
events (only relevant for an example!),
    # so you can easily see that `set_schedule` is working.
    start=time(second=0),
    end=time(minute=2),
    action=Action.charge if old_events and old_events[0].action ==
Action.export else Action.export,
    repeat=Days.Monday | Days.Wednesday | Days.Friday,
    charge_from=AdditionalSource.grid,
)
super_ems.set_schedule(site_id=site_id, events=[event])
```

Raw JSON

POST	/set/{site_id}/schedule
Set the schedule for the given site. Zero or up to 30 events can be set.	
PARAMETERS	
NAME	DESCRIPTION
site_id Required ***integer*** path	Unique site number to which the schedule is set.
{body} Required JSON body	Array of JSON objects that define the events, events that come earlier in the list have higher precedence. `[ { "start": "09:30", "end": "09:00", "action": "charge", "repeat": 64 } ]`
start Required ***string*** JSON structure	Start time in ISO 8601 format without seconds, "HH:MM". Event active start <= time < end. Note if a time is repeated due to daylight saving, the event is repeated. Similarly, if an hour is absent due to daylight saving the event won't run during the absent time.
stop Required ***string*** JSON structure	End time in ISO 8601 format without seconds, "HH:MM". Event active start <= time < end. Note if a time is repeated due to daylight saving, the event is repeated. Similarly, if an hour is absent due to daylight saving the event won't run during the absent time.

repeat Required ***integer*** JSON structure	It is the equivalent value of a byte. Think of a week as an array of days and for each day in the schedule is on (1) or off (0). With the 7th bit zero. Bit 6 is Saturday and Sunday is bit zero. Assign a value (1 or 0) to each day. If you convert the resulting week (byte) from binary to integer you will get the value for this field.
action Required ***string*** JSON structure	One of: "export" | "charge".

HEADERS

Key	Value
Authorization	'Bearer ' + access_token

RESPONSE

CODE	DESCRIPTION
200	Success
404	Not found. You may need to review the URI you are using on your request.
409	Client Error. You may need to review your request parameters.
500	Server Error. Please notify SuperEMS API team about this type of messages.

Log of Notifications for a Given Site

Python

The notification log for a site is obtained from a log call e.g.:

```
log = super_ems.log(
    site_id=site_id,
    start=now - timedelta(days=30),
    end=now,
)
```

The log call also allows filtering by type etc., which can be useful to reduce method response delay if a large period is requested (see source code for details of filtering).

The returned Log contains a list of Notifications:

```
@dataclass(frozen=True, order=True, kw_only=True)
class Notification(TimeStamp):
    """A notification in the log (has `id`, an initiator, a datetime, a
type, a code, and a description)`."""

    id: str
    """Unique ID string for the notification."""
    initiator: str
    """
    Identifies the agent or user who triggered the notification.
    Initiators have two different formats,
    an email (for registered SuperEMS users) or a Modbus ID (for devices
warnings or faults messages),
    depending on where the event was generated.
    All initiators are unique strings per site and can therefore be used
for tracking and auditing notifications.
    """
    type: Type
    """The type of the notification: fault, warning, or schedule."""
    code: int
    """Code from type category (see Appendix A for faults and B for
warnings, 200 for a schedule change)."""
    description: str
    """Description of the notification."""
```

Raw JSON

GET	/fetch/{site_id}/log
Retrieves the log of notifications associated to the given site, applying the requested filters.	
PARAMETERS	
NAME	DESCRIPTION
site_id Required ***integer*** path	Unique site number from which the log is requested.
start Required ***string*** query	Start date. ISO 8601 format, no time, nor time zone: "YYYY-mm-dd". Returned notifications for `start_no_fold <= record_date < end_no_fold` or `start_with_fold <= record_date < end_with_fold`.

end Required ***string*** query	End date. ISO 8601 format, no time, nor time zone: "YYYY-mm-dd". Returned records start_no_fold <= record_date < end_no_fold or start_with_fold <= record_date < end_with_fold.
type Optional ***string*** query	Notification type. One of the listed below: • fault: Refer to " • Appendix A: Faults". • warning: Refer to "Appendix B: Warnings". • info: Refer to "Appendix C: Information".
code Optional ***int*** query	Any valid code under the correspondent type category. Please refer to " Appendix A: Faults", "Appendix B: Warnings", and "Appendix C: Information".
initiator Optional ***string*** query	Identifies the agent or user who triggered the notification. Initiators have two different formats, an email (for registered SuperEMS users) or a Modbus ID (for devices warnings or faults messages), depending on where the event was generated. All initiators are unique strings per site and can therefore be used for tracking and auditing notifications.

HEADERS

Key	Value
Authorization	"Bearer " + access_token

RESPONSE

```
[
    {
        "id": "ABCD1234567890",
        "initiator": "user.name@company.com",
        "date": "2023-01-01 6:17:23",
        "type": "fault|warning|info",
        "code": "200",
        "description": "Generic description"
    },
    …
]
```

CODE	DESCRIPTION
200	Success
404	Not found. You may need to review the URI you are using on your request.
409	Client Error. You may need to review your request parameters.
500	Server Error. Please notify SuperEMS API team about this type of messages.

Appendix E: Connecting External Devices

ER supplied devices and additional user supplied devices can be connected to the superEMS™ secondary controller or superEMS™ main controller.

A list of approved user-supplied devices that can be connected to your superEMS™ secondary controller or superEMS™ main controller can be found here. If you require integration with another device that is not listed please contact Energy Renaissance on 1300 472 020, email service@energyrenaissance.com, or visit energyrenaissance.com/service to fill in a service and support form (charged for service).

Devices can be connected either via RS485 or TCP/IP. In your superEMS™ secondary RS485-1 is reserved for ER supplied components and RS485-2 is for you to connect your battery inverter or DC connected PV inverters (if required). Terminals on the superEMS™ main controller are also reserved for your external devices. For RS485 connections you MUST use 120 Ω instrumentation cable and ensure there is a terminating 120 Ω resistor on the last device in the daisy chain (do not use a multidrop – use a daisy chain). This termination can be checked by removing the cable from the RS485 terminals on the superEMS™ secondary controller and measuring the resistance over the two cables via a multimeter on ohms.

You must assign your devices different Modbus ID's, please refer to the device manual of that product for information on how to do this. If connecting via RS485 please ensure that all the devices connected to the one RS485 line have the same baud rate, data bits, parity and stop bits. Please follow the below ID convention. ER supplied devices come pre-programmed with the correct Modbus ID. Even if the device is connected via Modbus TCP (ethernet) it still needs a device ID (the superEMS™ server and hence clients use the ID to track faults and warnings).

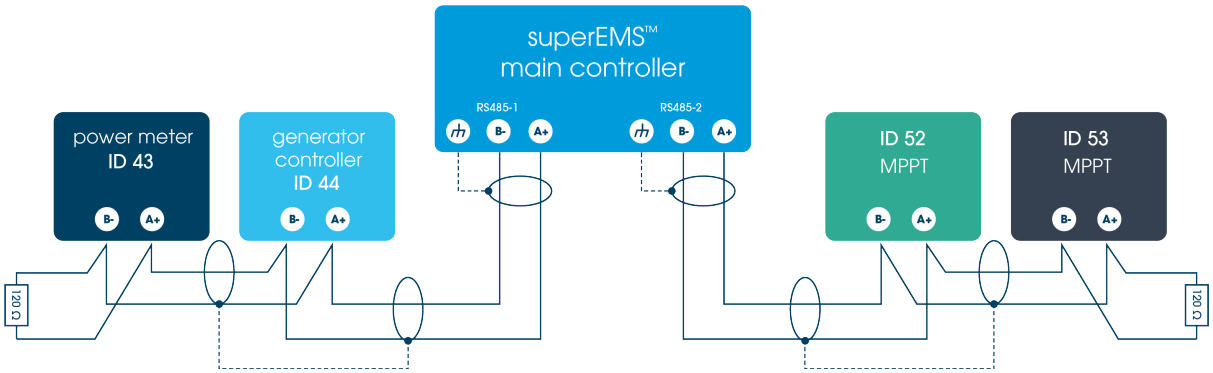
If your device is connected via TCPIP you will be required to change the static IP to be 172.29.189.XXX (X following the same ID convention of the Modbus ID)

If a site is reconfigured then the Modbus IDs will require reprogramming, contact Energy Renaissance on 1300 472 020, email service@energyrenaissance.com, or visit energyrenaissance.com/service to fill in a service and support form (charged for service).

Controller type and port	Modbus ID	Device
superEMS™ main controller		
RS485-1	145	Site EMS (Modbus option for superEMS™ main controller only)
RS485-2	43	Power meter
	44-50	Generator controller
	51	Automatic transfer switch
	52-82	AC connected PV inverters and/or loggers
superEMS™ secondary controller		
RS485-1	2-10	superEMS™ secondary
	27-42	ER supplied battery inverters
	87-95	ER supplied HVAC
	96-104	ER supplied relay module
	11-26	superRacks™
RS485-2	27-42	User supplied battery inverters (a.k.a. PCSs)
	83-113	DC/DC battery converters
	114-144	DC connected PV converters

Outdoor units have the RS485-2 connection of the superEMS™ secondary unit brought out to terminal blocks in the Point of attachment.

The above user configured communications and ID's must be communicated to Energy Renaissance prior to commissioning and the superEMS™ will be configured accordingly.



Connecting your external devices to your superEMS™ main controller via TCP/IP

Alternatively, to the above mentioned you can connect to your external devices via TCP/IP.

Your superEMS™ main controller comes with a network router and superEMS™ secondary controller with a network switch for connecting external devices.

Simply connect an ethernet cable to a port on the switch and to your device controller.

Generator on/off control

The superEMS™ main controller has the functionality to remotely operate generators fitted with approved remote modules. This functionality is provided by Modbus (RTU RS485 or TCP ethernet) communications connection. Ensure you read the generator controller manual to ensure it is configured to be remotely managed by an external device. See table above for Modbus ID range.

Appendix F: Black or initial start of your system (off-grid)

An initial or black start of your system requires AC voltage to energise the superEMS™ secondary controller and superEMS™ main controller so it can execute its program.

After installing your off grid superRack™ outdoor you will be required to follow the below steps to get your system running. Once the inverters are in off grid mode, they then will be able to support AC.

A black start describes an event where the batteries have discharged and the inverter has to shutdown to prevent battery over discharge In this case the superEMS™ main controller will instruct the generator to stay on and supply power to the batteries till they have recovered back up to at least 30% SoC.

The superEMS™ main controller has the intelligence to determine whether the system is in a black or initial start. In both cases please always follow the below steps. After completing the steps, you aren't required to do anything further.

1. Ensure that the internet connection is working.
2. Turn off the circuit breakers to your load.
3. Turn off your circuit breakers to your PV.
4. Ensure that the UPS that powers the logic supply in stage 11 below is charged and functioning.
5. Ensure that the breakers for the superBMS™, superEMS™ secondary controller, and superEMS™ main controller are all closed.
6. Ensure that the battery inverters' breakers are all closed.
7. Select manual mode on your generator.
8. Select off-grid (grid forming) mode on your generator.
9. Start your generator.
10. Wait for a text and/or email from the superEMS™ main controller to tell you to turn off the generator. This text could take a while (a few hours), it occurs when the SoC is above 30% and is therefore dependent on the initial SoC and the power of the generator and the size of the battery. Progress on charging the battery from the generator can be monitored using the superEMS™ UI client for the appropriate site.
11. Stop the generator. The battery inverter will take over and form the grid. A UPS is required to keep the logic supply up to the BMS and superEMS™ main and secondary during this period.
12. Switch your generator controller back to on grid (grid following) mode.
13. Switch your generator controller back to remote mode.
14. Turn the PV breakers back on.
15. Turn the load back on.