Appendix for Schneider Electric SunEzy inverters

WebdynSun

The monitoring gateway for your solar installation

CONTENTS

1 Connecting the gateway to Schneider Electric (SunEzy) inverters via the RS485(A) communications bus 3
1.1 Connecting at the end of the bus 3
1.2 Connecting in the middle of the bus 5
2 Discovering the Schneider Electric (SunEzy) inverter network and the file IDsite_daq.ini 7
3 Definition files for Schneider Electric (SunEzy) inverter families 8
4 Status variables 9
5 Schneider Electric (SunEzy) inverter alarm files 11
6 Schneider Electric (SunEzy) inverter parameter files 13
7 Schneider Electric (SunEzy) inverter parameter modification command files 13

1 Connecting the gateway to Schneider Electric (SunEzy) inverters via the RS485(A) communications bus

This chapter describes the connection of the WebdynSun to the RS485 communications bus for communication with Schneider Electric (SunEzy) inverters.
The gateway may be connected at the end of the RS485 communications bus or in the middle of the bus.
The gateway can handle up to 100 Schneider Electric (SunEzy) inverters.

⚠	▪ All wiring must be carried out only by a specialised qualified electrician. ▪ Prior to installation, all equipment connected to the corresponding communications bus must be disconnected on both sides (DC and AC). ▪ Respect all safety recommendations appearing in inverter documentation.

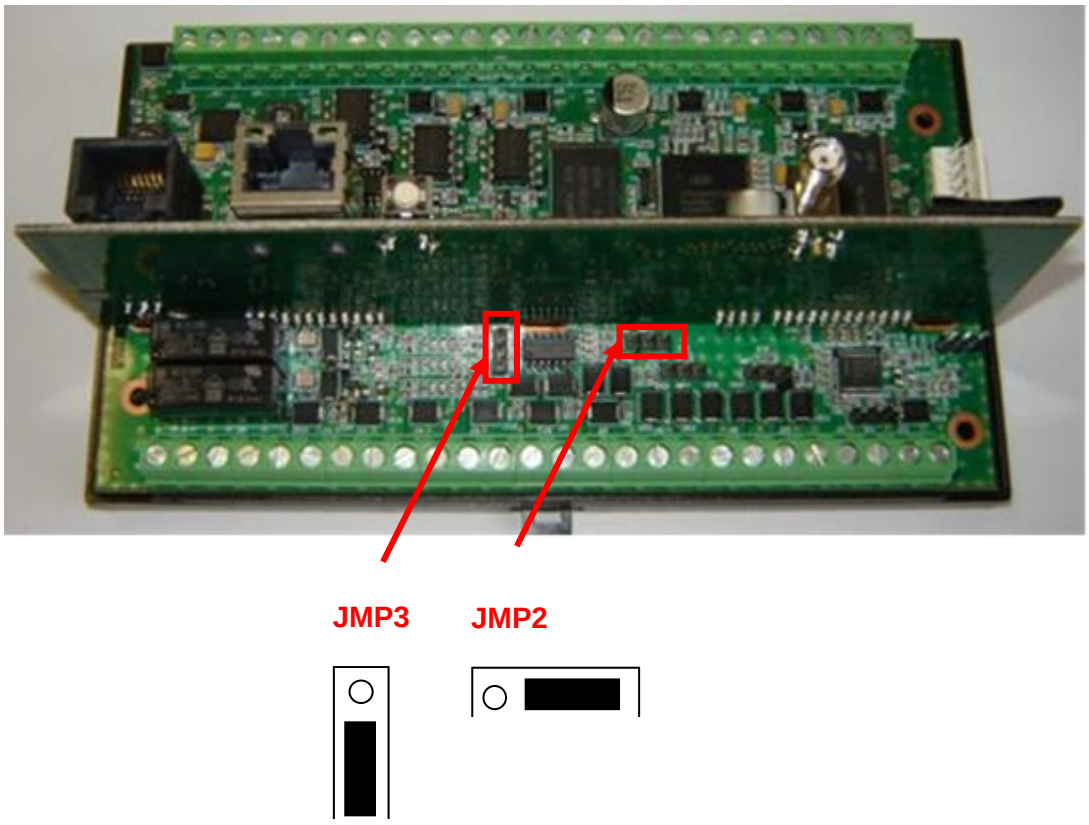
To ensure correct operation of the RS485 data bus, it must be fitted with a 120 ohm termination resistor at both ends.
Depending on the position of the gateway on the bus, this terminator plug must be activated or deactivated via a pair of jumpers mounted inside the casing.

1.1 Connecting at the end of the bus

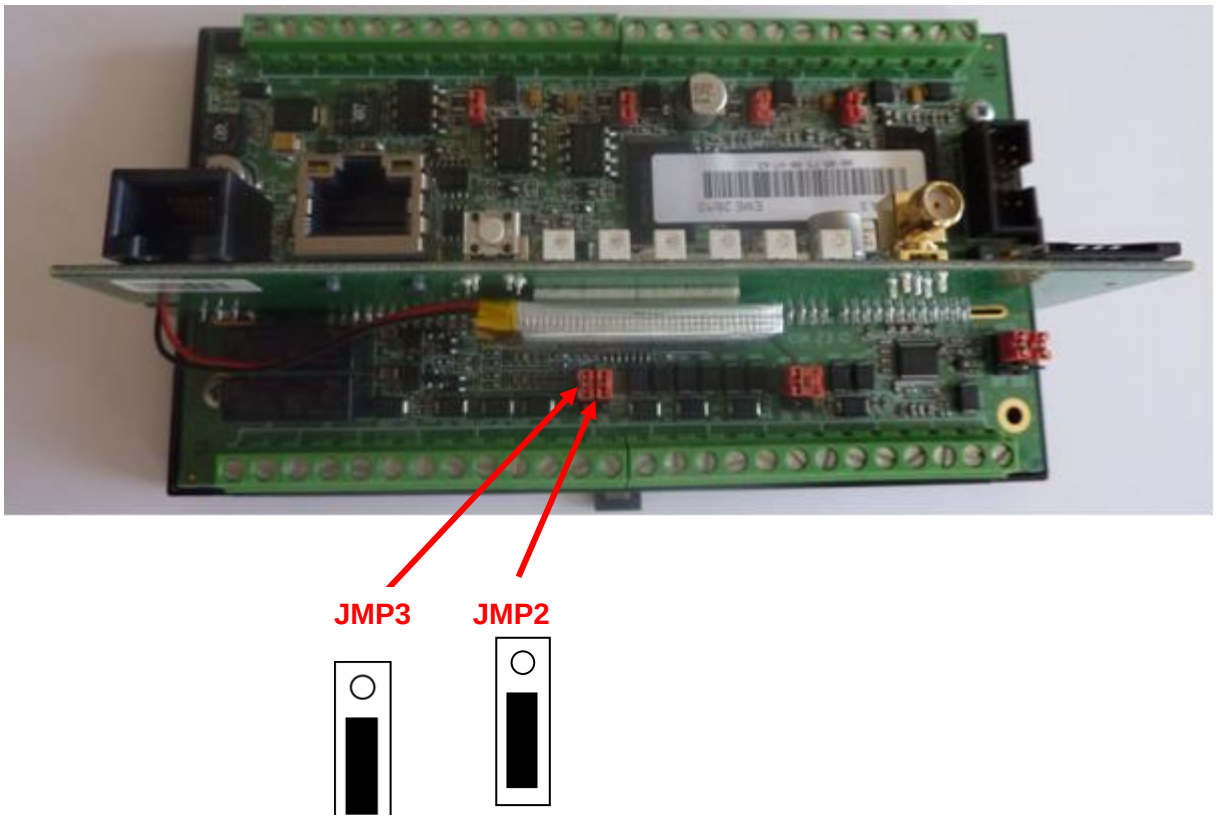
Positions of the JMP2 and JMP3 jumpers:
Only the JMP2 jumper inside the casing needs to be set to activate the 120 ohm bus terminator plug.

ℹ	By default, the gateway is configured to be located at the end of the bus. Consequently, the jumpers are plugged into the “Activation” position of the 120 ohm terminator.

Hardware version 1:



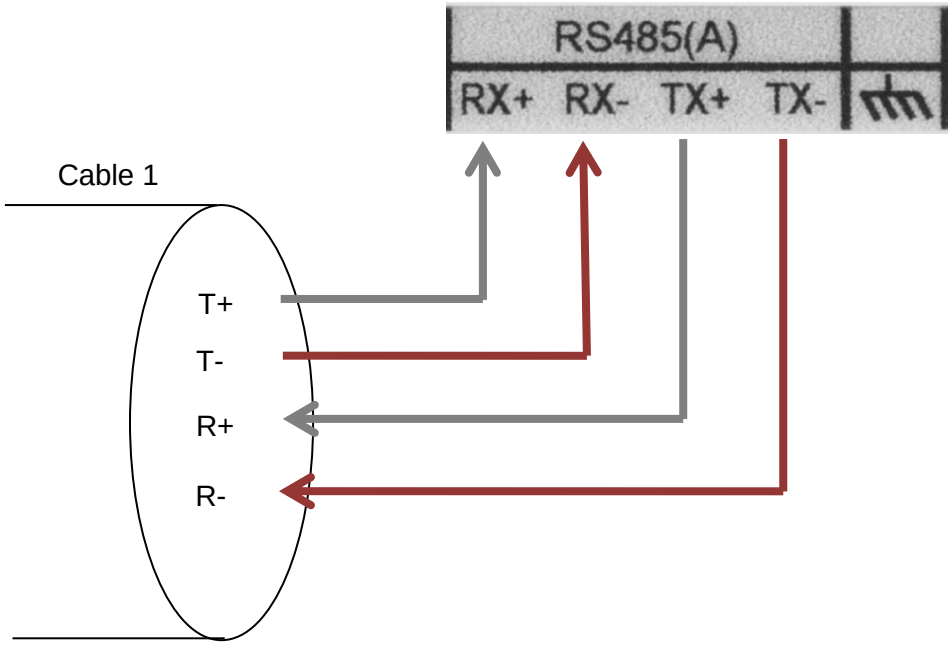
Hardware version 2:



Connecting inverters to the RS485 bus:
Consult the inverter documentation regarding the principles governing RS485 connection and wiring.

Once the RS485 cable has been brought up to the gateway:

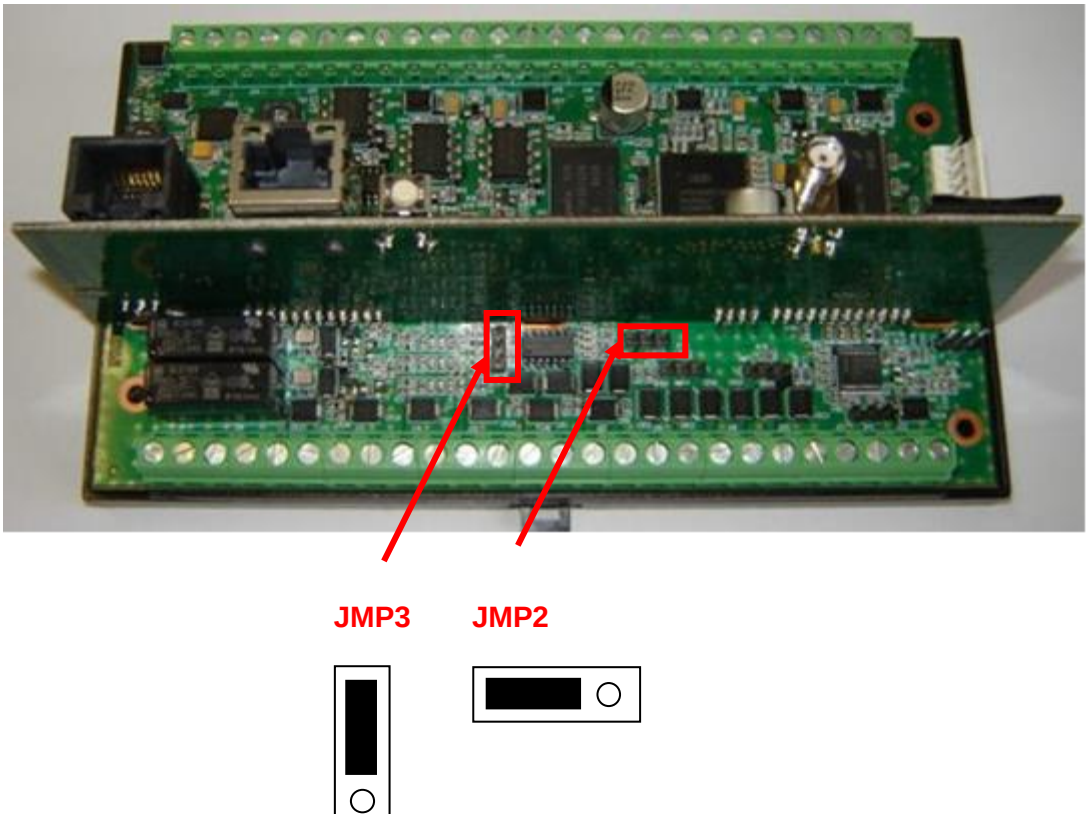
1. Strip around 4 cm of the outer sheath from the RS485 communications cable.
2. Trim the shielding back to match the cable sheath.
3. Bare around 6 mm of the wires.
4. Connect the conductors to the connector marked RS485(A), respecting the allocations in your RS485 communications bus.



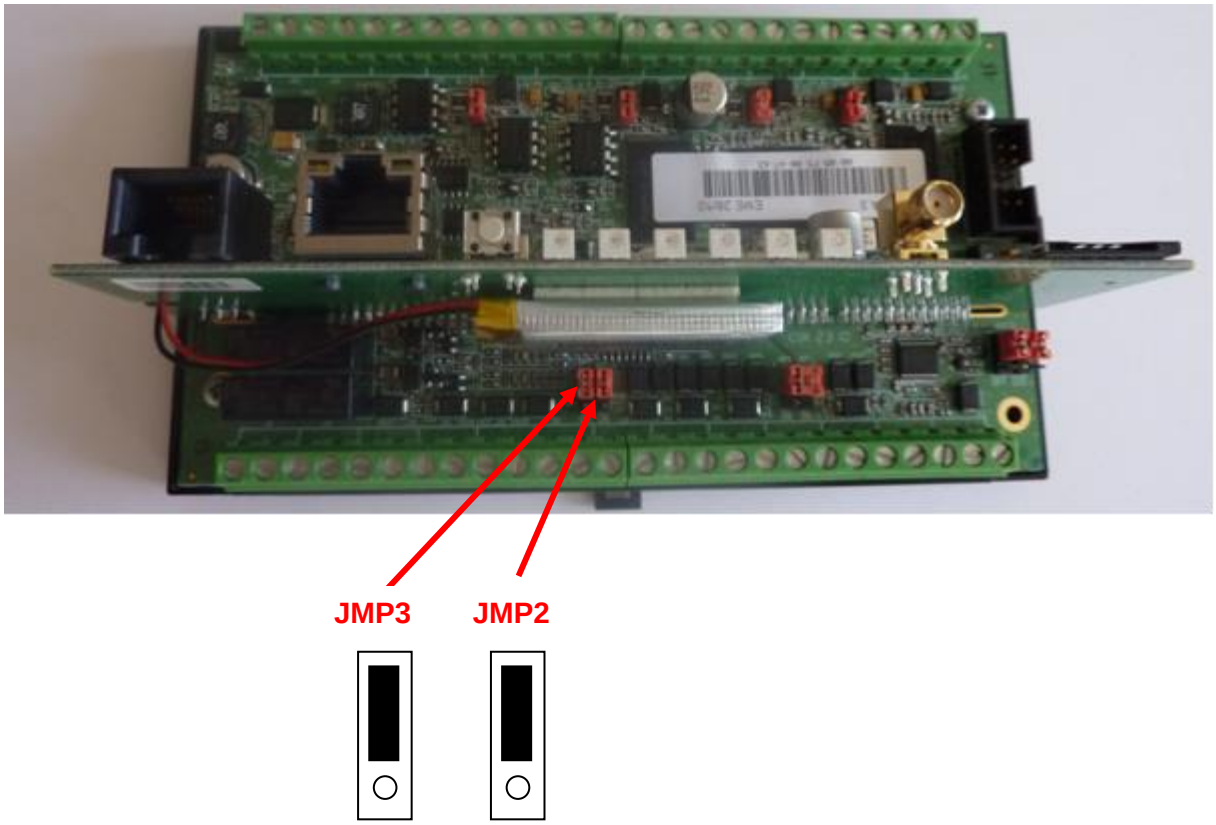
1.2 Connecting in the middle of the bus

Positions of the JMP2 and JMP3 jumpers:
Both the JMP2 and the JMP3 jumpers inside the casing must be positioned so as to deactivate the 120 ohm bus terminator plug.

Hardware version 1:

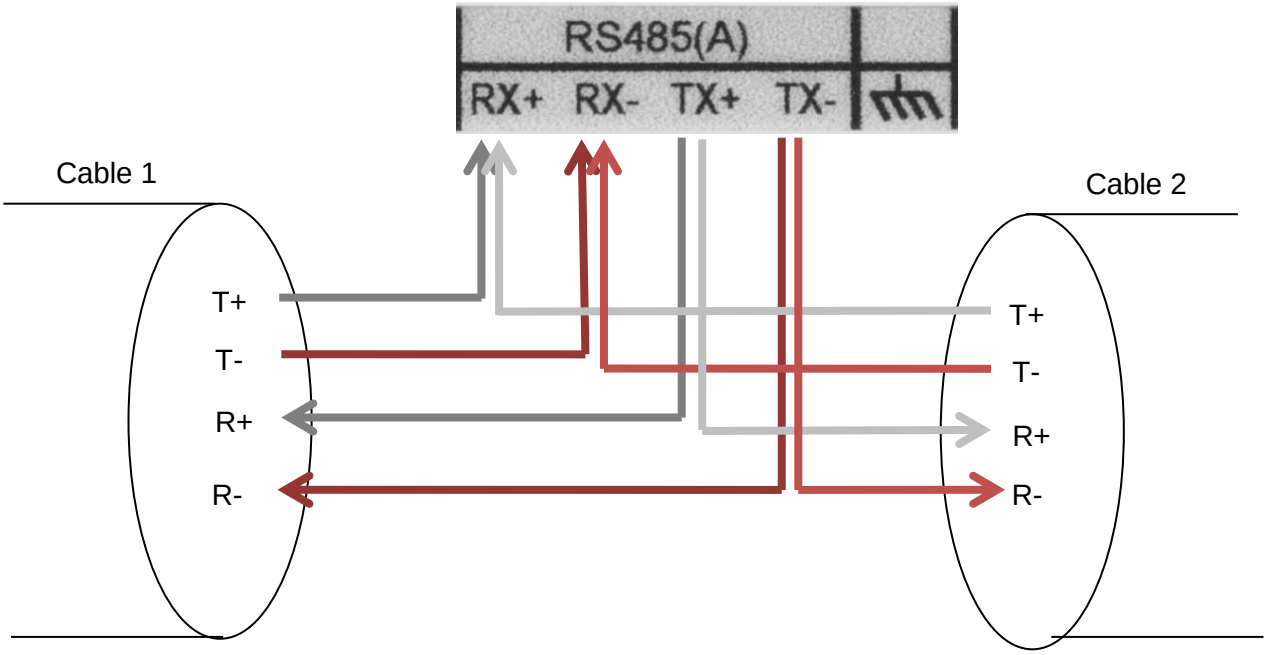


Hardware version 2:



Connecting inverters to the RS485 bus:
Consult the inverter documentation regarding the principles governing RS485 connection and wiring.
Once the two RS485 cables have been brought up to the gateway:

1. Strip around 4 cm of the outer sheath from the RS485 communications cable.
2. Trim the shielding back to match the cable sheath.
3. Bare around 6 mm of the wires.
4. Connect the conductors to the connector marked RS485(A), respecting the allocations in your RS485 communications bus.



2 Discovering the Schneider Electric (SunEzy) inverter network and the file IDsite_daq.ini

The SunEzy protocol allows for the discovery and addressing of inverters.

This discovery can be done via the built-in Web server or via a command file.

Following this discovery, if a connection is made and there is no *IDSite*_daq.ini file present in the /CONFIG directory of the FTP server, an *IDsite*_daq.ini file containing the declarations of all the inverters discovered will be generated automatically, then uploaded to the server.

A definition file corresponding to each inverter is generated during the discovery phase, and is used to provide information for the automatically generated *IDsite*_daq.ini file.
The definition files are also uploaded during the connection, if they are not already present on the server.

3 Definition files for Schneider Electric (SunEzy) inverter families

Filename format:
*IDSite*_ INV_*Family*.ini

This file is uploaded by the gateway after an inverter discovery phase. It may be modified by the user.
IDSite: site identification number.
Family: inverter family.

Example:

```
1;3;1;0;Phase number;-;0;0;1
2;3;2;1;VA rating;V;0;0;1
3;3;3;7;Firmware Ver.;-;0;0;1
4;3;4;12;Model name;-;0;0;1
5;3;5;28;Manufacturer;-;0;0;1
6;3;6;44;Serial Number;-;0;0;1
7;3;7;60;Nom_Vpv;V;0;0;1
8;4;0;0;Vpv-Start;0.1V;0;0;17
9;4;1;2;T-Start;s;0;0;17
10;4;4;4;Vac-Min;0.1V;0;0;17
11;4;5;6;Vac-Max;0.1V;0;0;17
12;4;6;8;Fac-Min;0.01Hz;0;0;17
13;4;7;10;Fac-Max;0.01Hz;0;0;17
14;4;8;12;Zac-Max;mOhm;0;0;17
15;4;9;14;Dzac;mOhm;0;0;17
16;2;0;0;Temperature;0.1 °C;0;0;2
17;2;13;2;E-Today;0.01kW.Hr;0;0;4
18;2;64;4;Vpv1;0.1V;0;0;2
19;2;65;6;Iac;0.1A;0;0;2
20;2;66;8;Vac;0.1V;0;0;2
21;2;67;10;Fac;0.01Hz;0;0;2
22;2;68;12;Pac;W;0;0;2
23;2;69;14;Zac;mOhm;0;0;2
24;2;71;16;E-Total;0.1 kW.Hr;0;0;4
25;2;73;20;h-Total;Hr;0;0;4
26;2;76;24;Mode;-;0;0;8
27;2;120;26;GVFaultValue;0.1V;0;0;1
28;2;121;28;GFFaultValue;0.01Hz;0;0;1
29;2;122;30;GZFaultValue;0.001Ohm;0;0;1
30;2;123;32;TmpFaultValue;0.1°C;0;0;1
31;2;124;34;PV1FaultValue;0.1V;0;0;1
32;2;125;36;GFCIFaultValue;0.001A;0;0;1
33;2;126;38;Error message;-;0;0;8
```

This file covers all the accessible data on the inverter, with:

Index;FunctionCode;DataCode;BufferIndex;Name;Units;A;B;Action

Details of the field “FunctionCode”:
Possible values:

- 3: Non-modifiable parameters;
- 4: Modifiable parameters;
- 2: Variables (3 types)
  - Cumulative: “E-Today”, “E-Total” and “h-Total”;
  - Status: “Mode” and “Error message”;
  - Analogue.

It is not necessary to understand the fields DataCode and Buffer Index to be able to operate the WebdynSun.
These fields enable the gateway to find the variable using the information from the SunEzy protocol.
The fields FunctionCode, DataCode and Buffer Index must under no circumstances be modified.

The action to be applied to each variable is specified at the end of the line:

- ***Action=0***: Variable not collected.
- ***Action=1***: Variable treated as read-only parameter.
- ***Action=17***: Variable treated as read-write parameter.
- ***Action=2***: Min/max/avg values.
- ***Action=4***: Instantaneous value.
- ***Action=8***: Variable treated as an alarm.

N.B. Depending on the variable type, several actions are applicable:

Variable type	Variable format	Possible actions
Instantaneous variables	Cumulative	0, 4
	Analogue	0, 2, 4
	Status	0, 4, 8
Parameters		0, 1,17

N.B. If the action defined is not consistent with the variable type, it is ignored.

4 Status variables

The following tables provide information on the codes for the Status variables.

This information is an extract from the Schneider Electric document:
SunEzy inverters communication protocol v2_3_1.pdf

Variable: Mode

(entry in the definition file: 26;2;76;24;Mode;-;0;0;8)

Mode		Description
Wait	0x00 0x00	PV voltage is between 70~130V and there isn't any fault .In the state there is not output power transmitted to grid voltage.
Normal	0x00 0x01	If PV voltage is over130V, the state changes to Normal state in Wait state. In the normal state, output power will be transmitted to grid voltage and MPPT calculation will be executed, bus voltage will be adjusted and output power as well as the sum of output power will be calculated.
Fault	0x00 0x02	Fault signal occurs: execute the protection steps, insulate the grid voltage to system, detect the grid voltage and fault and judge whether the fault has been removed. The state can be resumed automatically after the fault has been removed. The condition of entering Fault mode: 1.Temperature too high 2.Ground current too high 3.PV Voltage too high 4.Grid Voltage=0 5.Output impedance too high 6.Grid frequency out of range 7.Grid Voltage out of range 8.Isolation resistance too low
Permanent Fault	0x00 0x03	System Fault. The protection steps will be executed and auto restart disabled The condition of entering Permanent Fault mode: 1.Grid current DC offset 2.Grid current too high 3.Eeprom cannot be read or write in 4.Communication between CPU is fail 5.Bus Voltage too low or high 6.Compare measured value from two CPU 7.Main bridge relay fail

Variable: Error message:

(entry in the definition file: 33;2;126;38;Error message;-;0;0;8)

	Error message	Description
Bit31	Master-Slave-Fail	Communication between microcontrollers is failing
Bit30	EEPROM-Fail	EEPROM cannot be read or written
Bit29	Fac-Master-Fail	The master-frequency is out of tolerable range
~~Bit28~~	~~Fac-Slave-Fail~~	~~The slave frequency is out of tolerable range~~
Bit27	Zac-Master-fail	The master impedance is out of tolerable range
~~Bit26~~	~~Zac-Slave-fail~~	~~The slave impedance is out of tolerable range~~
Bit25	Rly1-Fail	Relay is Fail
~~Bit24~~	~~Rly2-Fail~~	~~Relay 2 Test Fail~~
Bit23	ENS-Vac-Fail	Different value between Master and Slave for grid voltage
Bit22	ENS-Fac-Fail	Different value between Master and Slave for grid frequency
Bit21	ENS-Zac-Fail	Different value between Master and Slave for grid impedance
Bit20	ENS-Mess-Fail	Different value between Master and Slave for dI, Fac, Uac or Zac
Bit19	Offset-Iac-Fail;	The DC injection check for grid Current is fail
Bit18	Zpv-PE-Fail	Isolation resistance of PV-plant out of tolerable range before connecting to the grid
Bit17	Vac-Master-Fail	Master-grid voltage measurement-value out of tolerable range
~~Bit16~~	~~Vac-Slave-Fail~~	~~Slave-grid voltage measurement-value out of tolerable range~~
Bit15	VpvMax-Fail	Pv input voltage is over the tolerable maximum value
Bit14	Test Fail	Auto Test failed
Bit13	Temperature -Fail	Over temperature fault
Bit12	Reserved	Reserve for active island detection
Bit11	Bus-Fail	Dc bus fault
Bit10	GFCI-Fail	Ground current is too high
Bit 9	No-Utility	Grid voltage =0
Bit8	Delta Z Fault	Delta GridZ Fault
Bit7	Device Fault	Device Fault
Bit6	Bus_High-Fail	Dc Bus voltage is too high.
Bit5	Bus –Low Fail	Dc Bus voltage is too low
Bit4	ENS-GFCI-Fault	Different value between Master and Slave for GFCI
Bit 3	ENS-DCI- Fault	Different value between Master and Slave for output DC current
Bit 2	Ref 2.5V Fault	The 2.5V reference inside are abnormal
Bit 1	DC Sensor Fault	The DC output sensor is abnormal
Bit 0	GFCI Failure	The GFCI detection circuit is abnormal

5 Schneider Electric (SunEzy) inverter alarm files

An alarm is generated, then transmitted to the remote server in the form of a file on the FTP server (*IDSite*_AL_*YYMMDD_hhmmss*.csv.gz).
The data stored is also uploaded at the same time as an alarm.

N.B. See 4. Status variables, for an explanation of the alarm codes.

Alarms of the Schneider Electric (SunEzy) type:
This type of alarm is raised on a change in the status of a variable defined as an alarm in the associated definition file.

Date - Time	Type	Field 1	Field 2	Field 3	Field 4

Where:

Date–Time	Timestamp when alarm triggered, in format YY/MM/DD-hh:mm:ss
Type	INV: Alarm type is Schneider Electric (SunEzy)
Field 1	*IDsite*_INV_*family*.ini: Name of reference definition file
Field 2	Serial number of the triggering Schneider Electric (SunEzy) inverter
Field 3	1 to N: Index of the triggering data item in the associated definition file.
Field 4	Status of the trigger item: 0 to n => Code corresponding to status messages or to messages indicating inverter malfunction.

Example:

```
09/12/10-08:45:38;INV;WEBDYN01_INV_ SunEzy_2000.ini;10224AH290;26;2
```

Mechanism for censoring alarms

For status variables configured as alarms, it is possible to specify, in the definition file, which values are to be censored, i.e. for which an alarm report should not be uploaded, even though the action code is 8.

This mechanism means that an alarm is not uploaded if the value taken by a variable with an action code of 8 has been declared in the list of values to be ignored.
If the status variable does not describe values but a bit field instead, the list of values to be ignored contains the bit values to be ignored. (0 to 7, 0 to 15 or 0 to 31 depending on the bit length of the variable.)

The list of values to be ignored is specific to each variable declared in the definition file with the action code Alarm (8), and is written in the field following that containing the Alarm action code.
The values to be censored are separated by the pipe symbol "|".

Example with the variable Mode: Alarm using coded values
The declaration line for this variable is in the definition file.

```
26;2;76;24;Mode;-;0;0;8;2|3
```

In this case, if the variable takes the value 2 (Fault) or 3 (Permanent Fault), the new value will be saved to the data file, but no alarm will be uploaded.

Example with the variable Error Message: Alarm from a bit field
The declaration line for this variable is in the definition file.

```
33;2;126;38;Error message;-;0;0;8;11|13
```

In this case, if bit 11 (Bus Fail) flips to 1, or if bit 13 (Temperature Fail) flips to 1, and no other bit changes, the new value of the whole field will indeed be saved to the data file, but no alarm will be uploaded.

6 Schneider Electric (SunEzy) inverter parameter files

Filename format:
*IDSite*_INV_P_*AAMMJJ_hhmmss*.csv.gz
Where:
IDSite: gateway identifier.

The parameters of the inverters present on the network are sent to the FTP server in two cases:

- During the installation phase triggered by pressing the push-button on the gateway;
- Via a command file of the type GATEWAY: Send inverter parameters (*see chapter on managing command files in the Operating Manual).*

The file format is as follows: (fields in green are optional data that can be enabled or disabled in *IDSite*_daq.ini (see *section 7.3* in the User's Manual)).

```
SNINV;10224AH290;1;
TypeINV;SunEzy_2000.ini;
21;1;2;3;4;5;6;7;8;9;10;11;12;13;14;15;27;28;29;30;31;32;
07/12/10-09:20:50;1;2000;S1.00;SunEzy 2000;PHOENIXTEC;10224AH290;
3600;1500;30;1920;2560;4751;5019;65535;65535;0;0;0;0;0;0;
SNINV;0123456789;2;
TypeINV;Pv-Inv_1000.ini;
27;1;2;3;4;5;6;7;8;9;10;11;12;13;14;15;16;29;30;31;32;33;40;41;47;48;54;55;
07/12/10-09:20:52;1;001000;01.00;Pv-Inv 1000;SCHNEIDER;0123456789;
3600;50;5;200;100;160;136;232;208;0;0;0;0;0;0;0;0;0;0;0;0;
```

For each inverter, this file details the variables defined as parameters in the file *IDSite*_INV_*family*.ini *(see Chapter 3: Definition files for Schneider Electric (SunEzy) inverter families))*.

SNINV = Serial number of inverter n (n=0 à 99 max).
TypeINV = Family of inverter n.

The column heading "1;2;3;4;5; … ; " is added if the option (DAQ_HeaderOption) is enabled in *IDSite*_daq.ini (see *section 7.3* in the User's Manual*).*

> To minimise the sizes of transmitted files, the parameter file is compressed in .gz format.

7 Schneider Electric (SunEzy) inverter parameter modification command files

Commands are transmitted to the gateway in the form of files uploaded to the FTP server (*IDSite*_CMD.csv). The command file is deleted from the server by the gateway after local

downloading. After execution of the commands, an acknowledgement file is sent to the server (*IDSite*_ACK_*AAMMJJ_hhmmss*.csv*)*.

The parameters of the commands depend on the type of command sent, as shown below:

Index	Type	Parameter 1	Parameter 2	Parameter 3

N.B. We must draw your attention to the risks associated with uncontrolled modification of inverter parameters. The consistency of the data sent out to the inverter must be checked on the server side before generating the command file.

Only variables defined as read-write parameters in the definition file can be modified (see Chapter 3: *Definition files for inverter families)*.

Index	1 to N: Unique Identifier enabling the command to be identified
Type	INV: Command of the Inverter type
Parameter 1	Inverter serial number
Parameter 2	Index of the parameter to be modified (first field of the definition file).
Parameter 3	Value of the parameter to be changed

It is advisable to request a report of all the inverter parameters after the update, for checking purposes.