SunSpec Energy Storage Models

SunSpec Alliance Interoperability Specification

Document #: 12032
Status: Draft
Version: 4

ABSTRACT

This document describes the SunSpec energy storage models.

Change History

D-1: Initial draft.

D-2: Added content related to the 801, 802 and 803 storage models.

D-3: Incorporated corrections and feedback from the working group.

D-4: Added flow battery models and detailed lithium-ion models based on workgroup discussions in 2015 and 2016. Consolidated 801 and 802 based on workgroup feedback. Enhanced battery state enumeration and associated flow diagram. Made multiple corrections and enhancements based on workgroup feedback.

Promoted models 802, 803, 804, 805 and 807 from DRAFT to TEST status. Incorporated working group feedback related to battery states, contactors, connection failure reasons and alarms. Corrected errors found in original Draft 4 release.

Editor

Andrew Miller, Doosan GridTech

Contributors

Matthew Bush, NEC Energy Solutions; John Corman, Nuvation; Bob Fox, SunSpec Alliance; Stefan Janhunen, Nuvation Energy; Joe Leland, Doosan GridTech; Rick Lyons, ViZn Energy; Joe McCabe, Energy Ideas; Ashutosh Misra, ITN Energy Systems; John Nunneley, SunSpec Alliance; Jake Parks, UniEnergy Technologies; Bill Randle, Advanced Energy; David Ridley, UniEnergy Technologies; Paul Siblerud, ViZn Energy; Tom Tansy, SunSpec Alliance; Vilayanur Viswanathan, Pacific Northwest National Laboratory

Introduction

This SunSpec Alliance Interoperability Specification describes the data models and MODBUS register mappings for storage devices used in stand-alone energy storage systems (ESS). The models in this specification may also be applied to photovoltaic systems with storage subsystems.

This specification is not specific to a single storage technology. The base models described herein are designed to support a variety of storage technologies such as lithium-ion batteries, vanadium redox flow batteries, advanced lead-acid batteries, and more. While an initial focus has been placed a small number of popular technologies (lithium-ion and redox flow batteries) it is expected that detailed models for other storage technologies will be added as the specification evolves.

For more information on the different types of energy storage technologies that are used in energy storage systems today, please see the *Energy Storage Technologies* page on the Energy Storage Association web site. You can find the page here:

http://energystorage.org/energy-storage/energy-storage-technologies

Additionally, the following IEC whitepaper contains additional information on electrical energy storage devices:

http://energystorage.org/system/files/resources/iecwp-energystorage-lr-en.pdf

References

[1] *IEC 61850 Basic Communication Structure - Distributed Energy Resources Logical Nodes*, IEC 61850-7-420, Edition 1.0

[2] Energy Storage Association (2014, October 1). *Energy Storage Technologies*. Retrieved from http://energystorage.org/energy-storage/energy-storage-technologies.

[3] *SunSpec Technology Overview*, SunSpec Document #12040. Retrieved from http://www.sunspec.org/download.

[4] *SunSpec Information Models*, SunSpec Document #12041. Retrieved from http://www.sunspec.org/download.

[5] *SunSpec Information Model Reference*. Retrieved from http://www.sunspec.org/download.

Relationship to Draft 3

The first publicly available draft of the SunSpec Energy Storage Models specification was published in the fall of 2014 and labeled "Draft 3". Draft 4 builds on this work and adds additional models to support flow batteries. This draft also corrects a number of issues in the earlier draft, and it incorporates other feedback from workgroup members. For a detailed comparison of the differences between Draft 3 and Draft 4, please see Appendix A.

Even though this specification is marked as a "Draft," the Energy Storage Workgroup believes that the information provided here may be use to implement communication interfaces in production systems.

IEC 61850 Information Model Alignment

The storage models in this specification have been designed to be in alignment with IEC 61850-7-420 wherever possible. To that end, the values in the models use similar names, units, behaviors, etc. to help the reader connect the communication interfaces defined here to the underlying 61850 information model.

Some portions of the storage models in this specification do not have an IEC 61850 equivalent. The SunSpec Storage Workgroup will work with the owners of IEC 61850-7-420 to see if that specification should be expanded to include these concepts.

Storage Overview

*Important Note: This specification assumes that you have downloaded and read the **SunSpec Technology Overview** and the **SunSpec Information Models** documents mentioned above. The storage models described herein build on core concepts which are described in these two SunSpec publications.*

This document describes a number of SunSpec models each with an identifier in the 800 series. An attempt has been made to design these models in a modular way so that they may be combined to address a variety of battery storage devices. The complete set of models is listed in **Table 1**.

Table 1: SunSpec Energy Storage Models

Model #	Name	Summary	Specification Status	Availability
802	Battery Base Model	Provides key monitoring and control points for all battery storage devices.	TEST	Draft 4
803	Lithium-ion Battery Bank Model	Provides monitoring and control points for a lithium-ion battery bank. Includes summary information for the strings within the bank.	TEST	Draft 4
804	Lithium-ion Battery String Model	Provides monitoring and control points for a lithium-ion battery string. Includes summary information for the modules within the string.	TEST	Draft 4
805	Lithium-ion Battery Module Model	Provides monitoring and control points for a lithium-ion battery module. Includes summary information for the cells within the module.	TEST	Draft 4
806	Flow Battery Bank Model	Provides monitoring and control points for a flow battery bank. Includes summary information for the strings within the bank.	N/A	Draft 5
807	Flow Battery String Model	Provides monitoring and control points for a flow battery string. Includes summary information for the modules within the string.	TEST	Draft 4

808	Flow Battery Module Model	Provides monitoring and control points for a flow battery module. Includes summary information for the stacks within the module.	N/A	Draft 5
809	Flow Battery Stack Model	Provides monitoring and control points for a flow battery stack. Includes summary information for the cells within the stack.	N/A	Draft 5

All SunSpec battery devices must implement the Battery Base Model (S 802). They may optionally implement one or more additional models specific to a battery storage technology (e.g. flow batteries).

The following top-level data elements are provided to describe each energy storage model:

- **C_SunSpec_ID** – A well-known value – 8xx that uniquely identifies this model as an energy storage model.
- **C_SunSpec_Length** – The length of the energy storage model in registers, not including the ID or the length registers.

The various device models are described in detail in the subsequent sections. All storage models, excepting the Battery Base Model, are optional, but if a particular storage model is used, all of the defined registers in that model must be present. Implementations should leave unused or unsupported data points within a storage model set to the “Not Implemented” value specified in the **SunSpec Technology Overview**. For example, the Not Implemented value for a 16-bit signed integer is 0x8000.

Settings are marked in the information model document with access RW. It is not required to support writable for all settings. Settings may be read-only if the setting is fixed or not settable via the communication interface. Such limitations shall be noted in the information model document.

Note that all of the storage models are padded to 64-bit boundaries to avoid register alignment issues. Additionally, padding has been used to ensure that all 32-bit values begin on a 32-bit boundary.

Repeating Blocks

Models S 803, S 804, S 805 and S 807 all make use of SunSpec's repeating block feature. For more information on repeating blocks, please see the **SunSpec Information Models** document.

The Model Length register in each of these storage models should be used to determine how many repeating elements exist in the Modbus register map. In each case, Model Length is a simple formula:

Model Length = Size of Fixed Block +

(Size of Repeating Block * Number of Repeating Elements)

To allow for future expansion in a given implementation, each of the storage models listed above also includes a count register which specifies how many of the repeating elements within the repeating range are actually implemented. This allows a vendor to reserve space in the register map for additional repeating elements that may be added later.

For example, a vendor who implements the Lithium-ion Battery Bank model (S 803) may use the Model Length register to indicate that there are 20 lithium-ion battery strings in the repeating portion of the model, while additionally setting the NStr register in S 803 to the value 15, which indicates that only 15 of the 20 battery strings are currently implemented. When a new string is added at a future date, only the NStr register needs to be updated.

Each of the spare elements in a repeating range should return the SunSpec Not Implemented value for all registers in the repeating block.

Vendor Models

A given vendor may find it necessary or desirable to expose additional information which is not covered by the storage models described here. This may be addressed by introducing one or more vendor models to the device definition. More information on vendor models can be found in the **SunSpec Information Models** document published by SunSpec.

Model Usage

The examples below illustrate how the models above may be combined to address different energy storage technologies and configurations. Note that the register maps in this section all assume that the register map begins at address 40001, even though the

SunSpec information model allows for register maps that begin at other addresses. See the **SunSpec Information Models** document for more information.

The examples provided here are not meant to be exhaustive. Configurations that combine the models in different ways are certainly possible.

Lithium-ion Configurations

Lithium-ion batteries are one of the most popular forms of energy storage. Part of the reason for their popularity is the flexibility of the technology. While a single lithium-ion module may be used in a residential energy storage application, multiple lithium-ion modules can be connected in series to create a larger energy storage system suitable for commercial and industrial applications. These strings can then be connected in parallel to form a grid scale energy storage system which might be deployed at a utility substation.

Table 2: Lithium-ion Battery Terminology

Term	Definition
Cell	A single energy or charge-storing unit
Module	A single enclosed unit consisting of a set of cells
String	Set of battery modules connected in series
Bank	Set of battery strings usually connected in parallel

The sections below describe how the information models in this specification may be used for different lithium-ion configurations.

Lithium-ion Battery Bank

For large applications such as grid connected energy storage systems, lithium-ion batteries are often arranged into a bank which may provide hundreds of kilowatt-hours to a few megawatt-hours of energy.

A lithium-ion battery manufacturer who wishes to implement a SunSpec-compliant communications interface for a battery bank must first implement the Common Model, which is a requirement for all SunSpec devices. The Common Model is defined in the **SunSpec Information Models** document listed in the References section above. The manufacturer must next implement the Battery Base Model (S 802) which provides basic information about the battery. Additionally, the Lithium-ion Battery Bank Model (S 803) should be implemented which provides information and control points specific to lithium-ion batteries.

In addition to providing information on the battery bank, model S 803 provides summary information for the strings within the bank using SunSpec's repeating block feature. This string information provides a key measurements and status values for each string, in addition to a select set of control points.

The communications interface for the battery bank should end with the End Model, which is another SunSpec requirement.

Table 3 below shows how the SunSpec models should be combined for this scenario. In this example it is assumed that there are 9 strings in the battery bank, which directly impacts the length of the S 803 model. This number will obviously vary by manufacturer and installation.

Table 3: Models Implemented for a Lithium-ion Battery Bank

Register	Description	Values	
4x40001	'SunS' Identifier	32-bit Identifier :	0x53756e53
4x40003	Common Model	Model ID : Model Length : ⋮	1 66
4x40071	Battery Base Model	Model ID : Model Length : ⋮	802 62
4x40135	Lithium-ion Battery Bank Model	Model ID : Model Length : ⋮	803 26 + (9 * 28) = 278
4x40415	End Model	Model ID : Model Length :	0xFFFF 0

Note that the register addresses in this table use the 4xNNNNN format where "4x" denotes "Read Holding Register" and NNNNN is the Modbus address.

Additional Lithium-ion String Detail

While the Lithium-ion Battery Bank Model (S 803) does provide summary information for the strings in the bank, in some installations it may be desirable to expose additional string or module information for monitoring purposes. The additional information may be useful in understanding how the battery bank is performing overall. The Lithium-ion Battery String Model (S 804) provides detailed string information in addition to summary information for the modules within the string. If this level of detail is desired, one instance of S 804 should be added to the register map for each string in the bank.

Additionally, the String Index (StrIdx) register in each S 804 model block should be used to identify the string within the bank.

Table 4 shows how the register map should be extended to incorporate one or more S 804 models. In this example it is assumed that there are 12 battery modules in each of the 9 battery strings.

Table 4: Additional Models for Lithium-ion String Detail

Register	Description	Values	
4x40001	'SunS' Identifier	32-bit Identifier :	0x53756e53
4x40003	Common Model	Model ID : Model Length : ⋮	1 66
4x40071	Battery Base Model	Model ID : Model Length : ⋮	802 62
4x40135	Lithium-ion Battery Bank Model	Model ID : Model Length : ⋮	803 26 + (9 * 28) = 278
4x40415	Lithium-ion Battery String Model (String #1)	Model ID : Model Length : String Index : ⋮	804 26 + (12 * 28) = 362 1
4x40779	Lithium-ion Battery String Model (String #2)	Model ID : Model Length : String Index : ⋮	804 26 + (12 *28) = 362 2
	(Additional S 804 Models for Strings 3 – 8 Go Here)		
4x42963	Lithium-ion Battery String Model (String #9)	Model ID : Model Length : String Index : ⋮	804 26 + (12 * 28) = 362 9
4x43327	End Model	Model ID : Model Length :	0xFFFF 0

Lithium-ion Battery String

In a commercial or industrial setting, a single series-connected set of battery modules may be arranged into a string to provide backup power, peak power limiting capability, or other functions.

In this scenario the Lithium-ion Battery String Model (S 804) should be combined with the Common Model and the End Model to form the communication interface for the battery string.

A Modbus register map for this scenario can be found in **Table 5**. In this example it is assumed that there is 1 string in the battery bank, and 5 modules in the string.

Table 5: Models Implemented for a Single Lithium-ion Battery String

Register	Description	Values
4x40001	'SunS' Identifier	32-bit Identifier : 0x53756e53
4x40003	Common Model	Model ID : 1 Model Length : 66 ⋮
4x40071	Battery Base Model	Model ID : 802 Model Length : 62 ⋮
4x40135	Lithium-ion Battery String Model (String #1)	Model ID : 804 Model Length: 38 + (5 * 16) = 118 String Index : 1 ⋮
4x40255	End Model	Model ID : 0xFFFF Model Length : 0

Note that the S 803 model is not included here as the bank model does not provide any additional information for a single lithium-ion string. However, including the S 803 model in the register map is certainly valid, and may result in increased compatibility with some Modbus masters.

Additional Lithium-ion Module Detail

While the Lithium-ion Battery String Model (S 804) does provide some information about the modules in the bank, in some installations it may be desirable to expose additional module or cell information for monitoring purposes. The additional information may be useful in understanding how the battery string is performing overall.

The Lithium-ion Battery Module Model (S 805) provides detailed module information in addition to summary information for the cells within the module. If this level of detail is desired, one instance of S 805 should be added to the register map for each module in the bank. Additionally, the Module Index register in each S 805 model block should be used to identify the module within the bank.

Table 6 shows how the register map should be extended to incorporate one or more S 805 models. In this example it is assumed that there is one string in the battery bank, five modules in the string, and 96 cells in each battery module.

Table 6: Additional Models for Lithium-ion Module Detail

Register	Description	Values	
4x40001	'SunS' Identifier	32-bit Identifier : ⋮	0x53756e53
4x40003	Common Model	Model ID : Model Length : ⋮	1 66
4x40071	Battery Base Model	Model ID : Model Length : ⋮	802 62
4x40135	Lithium-ion Battery String Model (String #1)	Model ID : Model Length: String Index: ⋮	804 38 + (5 * 16) = 118 1
4x40255	Lithium-ion Battery Module Model (Module #1)	Model ID : Model Length: Module Index : ⋮	805 42 + (96 * 4) = 426 1
4x40683	Lithium-ion Battery Module Model (Module #2)	Model ID : Model Length: Module Index : ⋮	805 42 + (96 * 4) = 426 2
	(Additional S 805 Models for Modules 3 & 4 Go Here)		
4x41967	Lithium-ion Battery Module Model (Module #5)	Model ID : Model Length: Module Index : ⋮	805 42 + (96 * 4) = 426 5
4x42395	End Model	Model ID : Model Length :	0xFFFF 0

Single Lithium-ion Battery Module

In residential or other behind-the-meter applications, a single lithium-ion battery module can be used to provide peak shifting and other services. The SunSpec Common Model, the Battery Base Model (S 802) and the Lithium-ion Battery Module Model (S 805) may be combined in this configuration to provide a complete communication interface to the battery. **Table 7** below shows how the models may be combined to form the complete register map.

In this example it is assumed that there are 96 cells within the battery module.

Table 7: Models Implemented for a Lithium-ion Battery Module

Register	Description	Values	
4x40001	'SunS' Identifier	32-bit Identifier :	0x53756e53
4x40003	Common Model	Model ID : Model Length : ⋮	1 66
4x40071	Battery Base Model	Model ID : Model Length : ⋮	802 62
4x40135	Lithium-ion Battery Module Model	Model ID : Model Length : Module Index : ⋮	805 42 + (96 * 4) = 426 1
4x40563	End Model	Model ID : Model Length :	0xFFFF 0

Lithium-ion Split Bus String

One battery connection topology that deserves special attention is a battery string or bank that has a center ground point. In this case, half of the battery modules in each string have a negative potential with respect to ground while the other half of the modules have a positive potential with respect to ground. The DC bus is split into a positive and negative half.

This connection scheme effectively halves the voltage potential measured from any point within the battery to ground. The battery is actually a three-terminal device in this case with a positive, a negative, and a ground terminal.

Batteries with split DC busses may be represented as battery banks with an even number of strings. Half of the strings are connected between ground and the positive terminal, while the rest are connected between ground and the negative terminal. Separating the positive and negative sub-strings of modules into separate strings within the battery model allows voltage, current, and SOC information to be exposed for each sub-string. This enables identification of imbalances within the overall system that would otherwise remain hidden.

Battery voltage and battery current in the battery base model (S 802) are always expressed with respect to the positive and negative battery terminals regardless of whether the battery uses a split DC bus or not. For split bus applications, battery voltage would actually be the sum of the negative and positive sub-string voltages. And overall battery current should equal both the negative and positive sub-string currents.

Table 8 below illustrates a hypothetical model layout for a battery with a single string that has a center ground point. This battery is actually represented as a battery bank with two strings, one for the positive sub-string of modules and the other for the negative sub-string of modules. Larger banks with multiple center grounded strings would follow this pattern.

In this example it is assumed that there are 5 modules per string.

Table 8: Models Implemented for a Split Bus String

Register	Description	Values
4x40001	'SunS' Identifier	32-bit Identifier : 0x53756e53
4x40003	Common Model	Model ID : 1 Model Length : 66 ⋮
4x40071	Battery Base Model	Model ID : 802 Model Length : 62 ⋮
4x40135	Lithium-ion Battery Bank Model	Model ID : 803 Model Length : 26 + (2 * 28) = 402 ⋮
4x40539	Lithium-ion Battery String Model (String #1 - positive sub-string of modules)	Model ID: 804 Model Length: 38 + (5 * 16) = 118 String Index: 1 ⋮
4x40659	Lithium-ion Battery String Model (String #2 - negative sub-string of modules)	Model ID: 804 Model Length: 38 + (5 * 16) = 118 String Index: 1 ⋮
4x40779	End Model	Model ID : 0xFFFF Model Length : 0

Modbus Address Space Considerations

If a manufacturer chooses to implement each of the lithium-ion models for a battery bank (S 802, S 803, S 804, and S 805), the combined models may require more than 65,535 registers which is the limit for a Modbus register map.

Manufacturers should consider whether or not all models are necessary. For example, in a grid-connected energy storage system, it may not be necessary to provide the cell-level detail that is specified in S 805.

If all of the models are required, manufacturers may choose multiple logical Modbus devices to expose the data. For example, a Modbus slave with Station ID = 100 could expose S 802, S 803 and S 804, while a separate Modbus slave with Station ID = 101 could expose the S 805 models. Of course, the SunSpec Common Model would have to be present in both register maps.

Flow Battery Configurations

Like lithium-ion batteries, flow batteries (e.g. vanadium redox flow batteries, zinc-bromine flow batteries, iron-chromium flow batteries, etc.) may be configured in multiple ways to address different scenarios. This specification uses the terms in **Table 9** below when describing these configurations.

Table 9: Flow Battery Terminology

Term	Definition
Cell	An electrochemical cell that reversibly converts chemical energy directly to electricity, and allows for a flowing electrolyte exchange with an external vessel
Stack	A group of connected cells, assembled in a mechanically contiguous form and usually connected in series electrically
Module	An assembly of one or more stacks that form a discrete unit and share electrical and electrolyte connections
String	Set of modules usually connected in series
Bank	Set of flow battery strings usually connected in parallel

Flow Battery String

Unlike lithium-ion batteries, grid scale energy storage systems based on flow batteries are often arranged in a single string. Because of their energy density, a single flow battery string is often able to provide several megawatt hours' worth of energy.

The Battery Base Model (S 802) and the Flow Battery String Model (S 807) form the core of the communication interface for flow battery strings. S 802 provides the basic battery information as it does in the lithium-ion case, while S 807 provides detailed information on the string and summary information on each of the modules in the string. **Table 10** below provides an example of S 802 and S 807 may be combined with the

Common Model and End Model in a flow battery installation. In this example it is assumed that there are 4 flow battery modules in the string.

Table 10: Models Implemented for a Single Flow Battery String

Register	Description	Values
4x40001	'SunS' Identifier	32-bit Identifier : 0x53756e53
4x40003	Common Model	Model ID : 1 Model Length : 66 ⋮
4x40071	Battery Base Model	Model ID : 802 Model Length : 62 ⋮
4x40135	Flow Battery String Model (String #1)	Model ID : 807 Model Length: 34 + (4 * 24) = 130 String Index : 1 ⋮
4x40267	End Model	Model ID : 0xFFFF Model Length : 0

Additional Flow Battery Configurations

While the single string configuration is common in the flow battery space, additional flow battery configurations are certainly possible. A future revision of this specification will define the S 806 model which may be used to expose the attributes of a flow battery bank (i.e. a collection of flow battery strings connected in parallel). Additionally, the Flow Battery Module Model (S 808) and Flow Battery Stack Model (S 809) will be fully defined so that these components may be exposed through the communication interface in a standardized way.

Battery Base Model (S 802)

The Battery Base Model (S 802) provides nameplate values, measurements and settings which are common to all batteries. This includes lithium-ion batteries, advanced lead-acid batteries, and flow batteries. In general a technology-specific model should be implemented in addition to Model 802 (e.g. Model 803 for lithium-ion batteries) but in cases where no specific support exists today, it is valid to implement Model 802 in isolation.

Nameplate Values

Nameplate values in the Battery Base Model allow an implementer to express the nameplate energy capacity of the device (WHRtg) in addition to nameplate charge and discharge rates (WChaRteMax and WDisChaRteMax).

For storage devices which have a measurable amount of self-discharge (i.e. decay), the DisChaRte field may be used to expose that quantity to a controller or other master.

State of Charge Management

Since all battery storage devices store a non-zero amount of energy, the Battery Base Model contains a number of values related to the state of charge of the storage device.

The State of Charge (SoC) value in the model expresses amount of usable charge remaining in the battery with respect to the actual capacity of the battery. This is a relative measure of remaining charge since the actual capacity of the battery will vary over time. When SoC is 100%, the battery is fully charged and cannot accept any more current during charge. When SoC is 0%, the battery is fully discharged and cannot provide any more current during discharge.

The absolute maximum amount of capacity in Ampere hours (Ah) that can be extracted from a battery may vary from the nameplate Ah capacity of the battery due capacity fade with usage and the age of the battery. The capacity may also vary due to imbalances within the battery itself.

A storage device manufacturer may want to limit a given device to a state of charge range that is less than 0% to 100%. For example, when some battery technologies are used in certain applications, it is not desirable to discharge the batteries to 0% as the lifetime of the batteries may be affected.

The Nameplate Max SoC (SoCMax) and Nameplate Min SoC (SoCMin) values in the Battery Base Model can be used to limit the usable state of charge range for a given storage device. These optional values are read-only as they are only intended to be set by the storage device manufacturer.

Should the state of charge on a storage device approach one of the nameplate limits, a warning may be issued by the device using the event flags on the Event 1 bitfield (Evt1). If the limit is then met or exceeded, an alarm in the same event field may be issued. For batteries which operate effectively over the entire SoC range, these warnings and alarms may not be useful and should be considered optional.

Application constraints on state of charge may be layered on top of any manufacturer constraints. For example, if a given storage device has a nameplate state of charge range between 10% and 90%, it may be desirable to further restrict the state of charge for a given application so that some amount of the energy capacity is held in reserve. The optional Max Reserve SoC (SocRsvMax) and Min Reserve SoC (SocRsvMin) settings are provided for this purpose. Note that while SocRsvMin may be the more common of the two settings (e.g. always reserve 30% of the energy in the battery for emergency situations) the SocRsvMax is also important. This allows a battery to reserve capacity for charging which might be useful in frequency response and other scenarios where emergency charging may be desired.

Figure 3 illustrates the different values and settings related to state of charge.

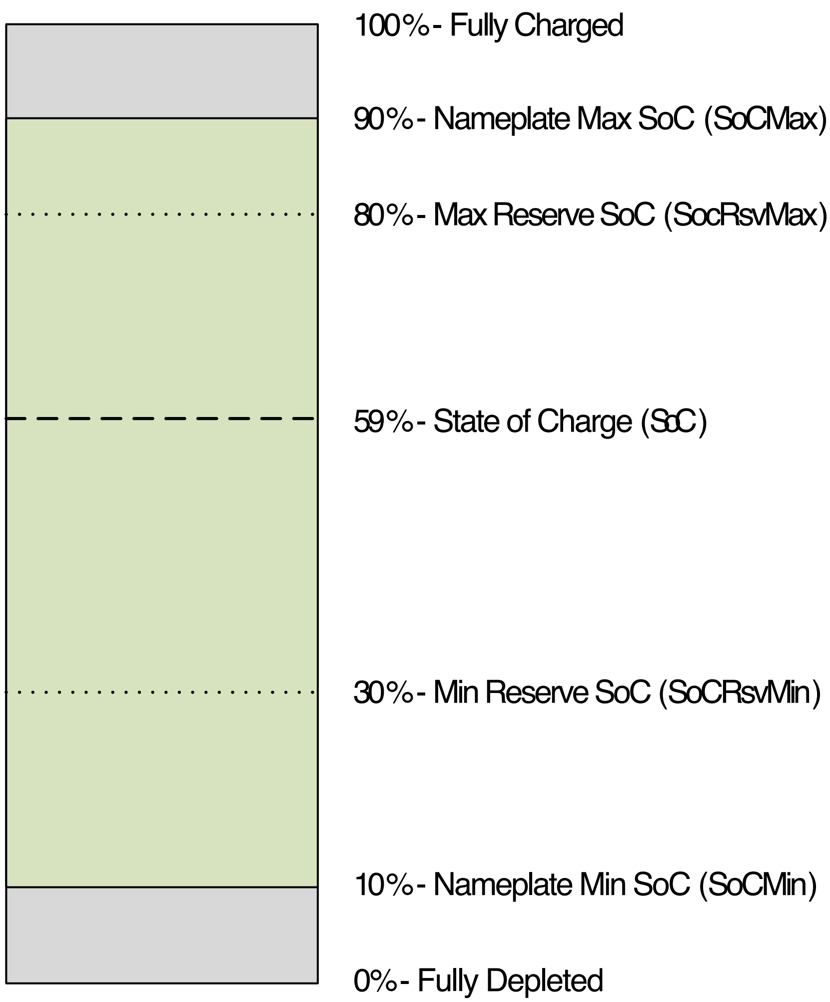

Figure 3: State of Charge Example

State of Charge and Maintenance Operations

Some battery types allow a portion of the battery to be taken offline. For example, the S 803 model allows a string within a lithium-ion battery to be disabled, which means that they will no longer connect when the battery bank is asked to connect. When a portion of a battery is disabled, that portion should be excluded from any state of charge calculation. Only enabled (i.e. online) portions of the battery should be included.

A meaningful SoC value should be reported regardless of the connected state of the battery. Even if the battery is disconnected, information about the energy stored in the battery is still useful. Depending on the battery technology multiple SoC calculations may be required to achieve this. For example, one calculation may be used when the battery is connected, and another calculation may be used when it is disconnected.

An SoC estimation based on cell or module voltage will be more accurate when the battery string is fully charged or discharged, disconnected or in the standby (low power) state. A connected battery string may use a current based SoC estimation from a known starting point to achieve the most accurate prediction.

Depth of Discharge

The optional Depth of Discharge (DoD) value in the model expresses the amount of charge that has been removed from the battery with respect to its rated nameplate capacity in Ampere hours. This is an absolute measure of charge since it is referenced against the fixed nameplate capacity.

When DoD is at 0% the battery is fully charged and cannot accept any more current. When DoD is at 100%, the full nameplate capacity of the battery has been discharged. Since the actual capacity of the battery varies over its lifetime, at 0% SoC, the DoD may be less than or greater than 100%

A given DoD always represents the same amount of charge flow out of the battery. The actual capacity of the battery is defined as the capacity in Ah when it is discharged from 100% SOC to 0% SOC. This represents the amount of charge that can be extracted from the battery when discharging from fully charged to fully discharged state.

State of Health and Cycle Counts

The health of the battery may be determined by inspecting the State of Health (SoH) value in the S 802 model. State of Health is defined as the ratio (expressed as a percentage) of the current rating of the battery with respect to the nameplate rating of the battery and end of life criteria. The rating of the battery may include one or more battery parameters that are selected as appropriate for the application and implementation.

When SoH is at 100%, the battery is performing at nameplate levels. This implies that all rating parameters are at least as good as the nameplate values. When SoH is at 0%, the battery has reached its end of life and requires service. If multiple rating parameters are used in calculating SoH, 0% implies that at least one rating parameter has reached its end of life threshold.

The Cycle Count (NCyc) register exposes the number of full discharge cycles that have been performed on the battery. One cycle corresponds to a full discharge to 100% DoD, followed by a full charge to 0% DoD. For many energy storage applications, full discharge cycles are not executed and instead the battery is only partially cycled. A given implementation is responsible for converting these partial cycles into full cycles which can be included in NCyc. However, the mechanism for combining these partial cycles is not specified here and left to the implementer.

Local vs. Remote Control

When maintenance is being performed on an energy storage device, remote control of the device should be prevented to ensure the safety of the personnel performing the maintenance.

The Control Mode value (LocRemCtl) in the Battery Base Model indicates whether or not remote control is allowed. Under normal conditions, this value is 0 which indicates that remote control is allowed. If local maintenance is required, on-site personnel will generally use a device-specific switch or HMI to put the storage device into local mode, at which point Control Mode will return 1 and all remote commands will be refused. Once the maintenance operation is complete, the same switch or HMI would be used to restore the ability to control the device remotely.

Heartbeats

Many of the storage devices being used today are large, complex systems made up of multiple subcomponents. A lithium-ion battery bank may be made up of multiple strings, each of which is made up of multiple modules, with everything fronted by an intelligent battery management system. Given this complexity, it is often desirable to not only to ensure that a valid communication channel exists, but also that the storage device is functioning at a basic level.

The Battery Heartbeat value (Hb) in S 802 is an unsigned numeric value which is incremented every second on the storage device. Periodically, this value resets to zero and the incrementing process continues (reset periodicity is up to the device manufacturer). A controller or other master can use this changing value to confirm that the energy storage device is healthy and able to provide updated values on demand.

Similarly, the Controller Heartbeat value (CtrlHb) in Model 802 can be used by the storage device to determine if it is properly communicating with the controller. If this value is not updated every second as expected, a storage device may choose to alter its state in some way, for example by entering into a standby or sleeping state.

The use of these heartbeat values is optional.

Battery Type and Alarm Information

The battery type enumeration (Typ) in the Battery Base Model is used to express the type of battery. An arbitrary Modbus master may inspect this value to learn what type of battery it is communicating with.

A battery device shall expose battery alarms and warnings may be exposed through the Battery Event 1 bitfield (Evt1). A wide array of standard alarms and warnings are included in the model, and provisions have been made to allow device-specific or manufacturer-specific alarms to be surfaced as well.

Latched alarms may be reset by writing a value of 1 to the Alarm Reset (AlmRst) register. When this value is received by the battery, the battery must attempt to reset all active alarms. Evt1 should be updated to reflect the result of this resetting. Once the battery is done resetting latched alarms, it must set Alarm Reset back to 0. This allows a master to wait while the reset is in progress, and when complete move on to other tasks.

Current, Voltage and Power Monitoring

Many of the values exposed on the Battery Base Model provide data which is useful in monitoring the health and status of the battery bank.

To allow masters to monitor the voltage, current and power for the battery bank, Model S 802 exposes the External Battery Voltage (V), Total DC Current (A) and Total Power (W) registers. Each of these registers exposes a measurement which may be used to ensure that the battery is operating as expected.

Given the importance of maintaining consistent voltage levels throughout the battery bank, the Max Cell Voltage (CellVMax) and Min Cell Voltage (CellVMin) values return the maximum and minimum voltages for all cells in the bank.

To help operators determine where these minimum and maximum values were measured, a battery may expose location information through the optional Max Cell Voltage String (CellVMaxStr), Max Cell Voltage Module (CellVMaxMod), Min Cell Voltage String (CellVMinStr) and Min Cell Voltage Module (CellVMinMod) registers. If implemented, these registers should report the index of the string and module in which the maximum or minimum voltage was measured.

Current and Voltage Limits

The Battery Base Model also provides values which express the instantaneous[1] charge and discharge current limits (AChaMax and ADisChaMax, respectively), and instantaneous maximum and minimum voltage limits (VMax and VMin). These values complement the nameplate ratings found in model 802, and allow a battery manufacturer to limit the current and voltage dynamically as the state of the battery changes. It is expected that battery controllers will monitor these values and ensure that charging and discharging operations fall within the maximums and minimums expressed in this model. Note that these current and voltage limits are not meant to be a protection mechanism—it is expected that the appropriate protection exists in the battery system and that there is no way for an energy storage system controller or inverter to circumvent these protection

---

[1] Note that these current and voltage limits are meant to be updated regularly, but based on data acquisition rates and other factors, how frequently they are updated will vary across implementations.

mechanisms. However, these current and voltage limits are important in that they communicate the bounds of acceptable operation. When properly honored, the batteries current and voltage protection mechanisms should never be triggered.

Note that AChaMax, ADisChaMax, Vmax and VMin are all marked as optional. In practice at least one pair (either AChaMax and ADisChaMax or Vmax and VMin) must be implemented.

For proper operation, a battery may need to know the current state of the connected inverter. The Set Inverter State (SetInvState) setting is used to provide this state information to the battery. A controller or other master should ensure that the current state of the inverter is written to this setting as soon as an inverter state change is detected.

Battery States

While batteries are in many ways passive devices, most provide a limited set of commands (e.g. a lithium-ion battery bank may offer the ability to connect and disconnect the battery strings). A controller or other master may execute one of these commands by using the Set Operation (SetOp) enumeration.

When a command like connect or disconnect is executed in the battery, the battery will transition from one state to another. The Battery State (State) value in model 802 expresses the current state of the battery. For example, if a controller uses SetOp to ask a disconnected battery to connect, a compliant battery will transition from the Disconnected state to the Initializing state, and then from the Initializing state to the Connected state.

It is worth noting that while it is initializing, a battery may wish to perform certain operations such as string SoC balancing to ensure that all strings have a similar SoC before the battery bank transitions to Connected. These operations may require the import or export of power. To allow for these operations, a battery may set Inverter State Request (ReqInvState) and Battery Power Request (ReqW) to ask the inverter to charge or discharge power. An energy storage system controller should monitor these values when a battery is in the Initializing state, and if power is requested, the connected inverter should be instructed to charge or discharge accordingly. Obviously system operating limits need to be respected in this scenario, so the controller is not required to honor the full magnitude of the battery request.

If a critical alarm condition is detected during the normal operation of the battery, the battery should raise the appropriate alarm(s) and then enter the Fault state. The contactors on the battery may or may not be opened upon transitioning to this state. Issuing an Alarm Reset (AlmRst) on a battery in the Fault state will move the battery back to the Disconnected state at which point the contactors should be open.

Some batteries may choose to provide protection when state of charge drops to a low level. When the battery enters this state, it should set the State register to SOC Protection so that the master realizes that battery operations may be limited.

Similarly, a battery may optionally choose to enter the Standby state if an extended period of inactivity is detected. Masters should be aware that even though the battery is expected to exit the Standby state automatically, transitioning from Standby to Connected may not be instantaneous, depending on the battery technology.

The state diagram in Figure 1 depicts the various battery states and the decision points which lead from state to state.

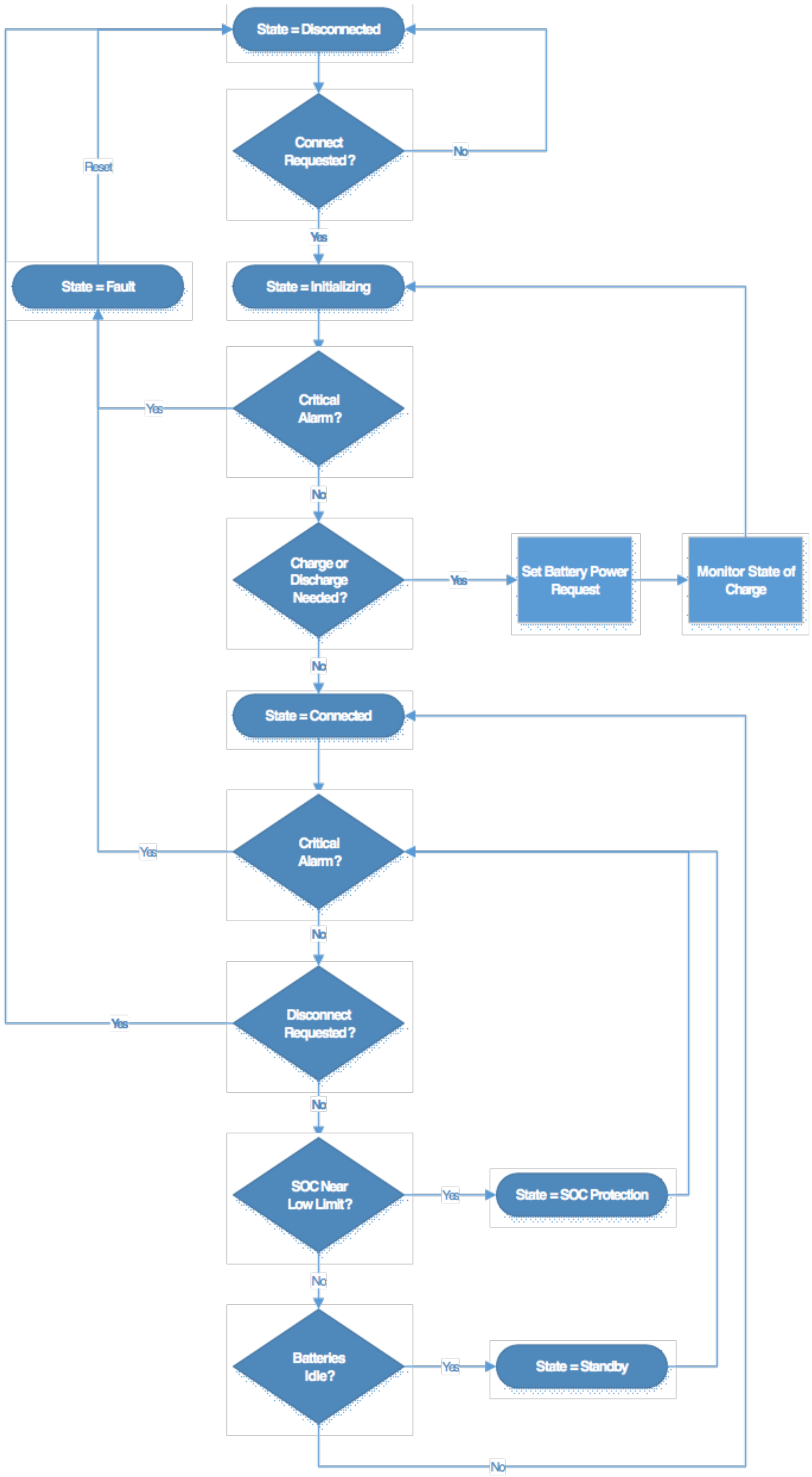

Figure 1: Battery State Diagram

Lithium-Ion Battery Bank Model (S 803)

The Lithium-ion Battery Bank Model exposes the unique characteristics of lithium-ion battery banks.

Monitoring Information

Given the importance of operating lithium-ion batteries at the right temperature, the Max Module Temperature (ModTmpMax) and Min Module Temperature (ModTmpMin) in S 803 return the maximum and minimum temperatures for all modules[2] in the bank.

To help operators determine where these minimum and maximum values were measured, a lithium-ion battery may expose location information through the optional Max Module Temperature String (ModTmpMaxStr), Max Module Temperature Module (ModTmpMaxMod), Min Module Temperature String (ModTmpMinStr) and Min Module Temperature Module (ModTmpMinMod) registers. If implemented, these registers should report the index of the string and module in which the maximum or minimum temperature was measured.

In addition to providing temperature data for the lithium-ion battery bank, Model S 803 also exposes key string measurements for voltage and current. The Max String Voltage (StrVMax) and Min String Voltage (StrVMin) registers provide the maximum and minimum measured voltage for all strings in the bank, while the Max String Current (StrAMax) and Min String Current (StrAMin) registers provide the maximum and minimum measured current for all strings in the bank.

Battery String Repeating Block

As mentioned above in the Terminology section, a lithium-ion battery bank is made up of one or more battery strings. Accordingly, it is quite often necessary to monitor and control the individual strings in the bank. Model S 803 includes a SunSpec repeating block which is repeated once for every string in the bank. See the **Repeating Blocks** section above for more information on how repeating blocks are used within the storage models.

The model size register in S 803 may be used to determine how many elements exist in the battery string repeating block. The String Count (NStr) field in S 803 indicates how many of those repeating instances correspond to physical strings.

Each string exposes a set of values which are similar to those that exist in S 802 and the S 803 fixed block. For example, String SoC (StrSoC), String SoH (StrSoH), String Max Cell Voltage (StrCellVMax), String Min Cell Voltage (StrCellVMin), String Max Module

[2] While many battery systems measure voltages at the cell level, it is common for temperatures to only be measured at the module level.

Temperature (StrModTmpMax), and String Min Module Temperature (StrModTmpMin) are all identical to their bank-level counterparts, but scoped to a given string.

A small number of string-specific values are also included in the repeating block. Module Count (StrNMod) provides a count on the number of battery modules in the string. Connection Failure Reason (StrConFail) is used to indicate why a given string failed to connect when the battery bank was last asked to connect. The String Status attribute (StrSt) provides useful information regarding the connected state of the string. StrSt also indicates whether or not a given string is enabled.

A battery string may be disabled for a variety of reasons including a fault condition, maintenance activity, or reduction in operational capacity. Additionally, strings can be disabled manually (i.e. by an operator) or automatically by the battery management system when certain conditions arise. StrDisRsn is used to convey additional context about why a particular string has been disabled. It can also be used to help distinguish strings that are potentially available for reconnection from strings that are completely unavailable.

The valid values for StrDisRsn are shown in **Table 11**.

Table 11: String Disabled Reasons

Value	Description
NONE	The string is enabled.
FAULT	The string encountered a fault condition during operation and was automatically disabled. It may be re-enabled by the controller once the fault condition has been cleared.
MAINTENANCE	The battery is undergoing maintenance. This string cannot be enabled and should be considered locked and tagged out.
EXTERNAL	An operator or an external controller has intentionally disabled the string to remove it from operation.
OTHER	The string is disabled for some other reason.

The Enable/Disable String (StrSetEna) setting allows a given string to be enabled or disabled by a controller or other master. A disabled string will not attempt to connect the next time that the battery is asked to connect. This provides a convenient mechanism to performance maintenance on a given string, while continuing to use the rest of the battery bank. Because disabled strings are excluded from bank-level calculations like State of Charge, battery manufacturers should disconnect the string at the time of disabling if possible.

Writing a 1 (ENABLE_STRING) to the StrSetEna register causes the string to be enabled. Writing a 2 (DISABLE_STRING) to the register causes the string to be disabled. When a string is disabled using StrSetEna, StrDisRsn should be set to the EXTERNAL value shown in **Table 11**.

When read, the StrSetEna register should reflect the current state in the enabling process, where 1 indicates that the string is in the process of being enabled, 2 indicates that the string is in the process of being disabled, and 0 indicates that the process is complete.

StrSetEna is optional, as the physical capability to connect or disconnect a specific string may not be implemented in a given system. If the Enable/Disable functionality is not supported, StrSetEna should return the SunSpec "Not Implemented" value for an enum16 (0xFFFF). However, it should be noted that this capability is very desirable, and all vendors are encouraged to implement StrSetEna if possible.

The Connect/Disconnect String (StrSetCon) writeable register in the battery string repeating block may be used to connect or disconnect an individual string. When used to connect a string, this function must honor the enabled state of the string, as specified by StrSetEna. For example, if a string is asked to connect using StrSetCon but that string is not enabled, the operation should fail and the CONTACTOR_STATUS bit in the StrStatus register should not be set.

Writing a 1 (CONNECT_STRING) to the StrSetCon register causes the string to be connected. Writing a 2 (DISCONNECT_STRING) to the register causes the string to be disabled.

When read, the StrSetCon register should reflect the current state in the connect/disconnect process, where 1 indicates that the string is in the process of being connected, 2 indicates that the string is in the process of being disconnected, and 0 indicates that the process is complete.

Lithium-ion Battery String Model (S 804)

The Lithium-ion Battery String Model exposes detailed information for a string within a lithium-ion battery bank. The fixed portion of the model contains a rich set of measurements and control registers for the string itself, and the repeating portion contains summary information for each of the modules within the string.

Fixed Block

The fixed block of S 804 builds on the repeating block of S 803 and adds additional detail. The one-based string index (Idx) is used to identify the location of the string within the battery bank.

The String Current (A) register in S 804 exposes the measured current for the string and the String Voltage (V) register exposes the measured voltage. The String SoC (SoC) register can be used to inspect the string's current state of charge, and the optional String Depth of Discharge (DoD) provides the amount of charge that has been removed from the battery string with respect to its rated nameplate capacity in Ampere hours.

The String Cell Balancing Count (NCellBal) expresses the instantaneous number of cells that are currently being charged or discharged through the use of active or passive balancing circuitry within the string.

Health information for the string is provided in the String SoH (SoH) and String Cycle Count (NCyc) registers. SoH exposes the health of the string as a percentage, whereas NCyc provides an indication of the number of full cycles that the string has executed. NCyc may be especially helpful when comparing multiple strings within the bank, particularly if modules of different ages are present within the bank.

The minimum and maximum cell voltage, temperature, state of health, and state of charge are present for the string and behave similar to the registers in S 802 and S 803.

Module Repeating Block

To enable the monitoring of the individual modules in the string, Model S 804 includes a SunSpec repeating block which is repeated once for every module in the string. For more information on how repeating blocks are handled in the storage models, please see the **Repeating Blocks** section above.

The model size register in S 804 may be used to determine how many elements exist in the module repeating block. The Module Count (NMod) field in S 804 indicates how many of those repeating instances correspond to physical modules.

Within the repeating block, the Module Cell Count (ModNCell) may be read to determine how many cell are present within the module. The health of the module is provided by the Module SoH (ModSoH) register.

Aggregate cell voltage data for the module is exposed through Max Cell Voltage (ModCellVMax), Min Cell Voltage (ModCellVMin) and Average Cell Voltage (ModCellVAvg). The Max Cell Voltage Cell (ModCellVMaxCell) and Min Cell Voltage Cell (ModCellVMinCell) location registers may be used to determine where the maximum and minimum values were read, respectively.

Cell temperature information is exposed in a similar manner through the Max Cell Temperature (ModCellTmpMax), Min Cell Temperature (ModCellTmpMin) and Average Cell Temperature (ModCellTmpAvg) registers, along with the Max Cell Temperature Cell (ModCellTmpMaxCell) and Min Cell Temperature Cell (ModCellTmpMinCell) location registers. It should be noted that if a battery does not measure cell temperatures for all cells in the module, interpolation may be used to calculate per-cell temperatures.

Lithium-ion Battery Module Model (S 805)

The Lithium-ion Battery Module Model exposes voltage, temperature, state of charge, and state of health at the cell level. It is the only model that exposes individual cell-level measurements and parameters rather than aggregate minimum and maximum values.

Fixed Block

The fixed block of S 805 builds on the repeating block of S 804 and adds additional detail. The Module Index (Idx) identifies the module by location within the battery bank. The module Serial Number (SN) is exposed to enable tracking of specific battery modules within the bank and to support service use cases.

The Balanced Cell Count (NCellBal) attribute exposes the number of cells that are currently being charged or discharged through the use of active or passive balancing circuitry within the module.

The module cycle count (NCyc) is provided as an indication of how much of the modules's cycle life has been used and can be used for comparison against other modules within the string and bank.

Aggregate cell voltage and temperature data for the module is exposed in a manner identical to the module repeating block in S 804.

Cell Repeating Block

A small number of registers has been provided in the repeating block of S 805 to allow the monitoring of the cells within the module. Cell Voltage (CellV), Cell Temperature (CellTmp) and Cell Status (CellSt) provide temperature, voltage and status information for each of the cells in the module.

Flow Battery Bank Model (S 806)

The Flow Battery Bank model will be fully defined in the next draft of this specification (Draft 5)

Flow Battery String Model (S 807)

The Flow Battery String Model has been designed to expose the unique characteristics of flow battery strings.

Monitoring Information

A primary function of the S 807 model is to provide monitoring information for a flow battery string.

At the top of the model two attributes are provided which provide information on the flow battery modules within the string. The Module Count (NMod) attribute indicates the total number of modules in the string, and the Connected Module Count (NModCon) attribute indicates how many of those modules are connected electrically. By definition, NModCon <= NMod.

Given the importance of maintaining healthy module and cell voltages within the string, a number of attributes are provided which expose detailed voltage information. The Max Module Voltage (ModVMax) and Min Module Voltage (ModVMin) provide the measured minimum and maximum voltages for all modules in the string. The optional Max Module Voltage Module (ModVMaxMod) and Min Module Voltage Module (ModVMinMod) provide location information which indicate where the maximum and minimum values were registered. Similarly, the Max Cell Voltage (CellVMax) and Min Cell Voltage (CellVMin) fields indicate the maximum and minimum cell voltages for the string, and the Max Cell Voltage Module (CellVMaxMod), Max Cell Voltage Stack (CellVMaxStk), Min Cell Voltage Module (CellVMinMod) and Min Cell Voltage Stack (CellVMinStk) optionally provide location information for these measured cell voltages.

The Flow Battery String Model also provides temperature data via the Max Temperature (TmpMax) and Min Temperature (TmpMin) attributes. These extremes are generated based on measurements taken from the electrolyte containers within the battery modules.

The model also exposes alarms and warnings in a way that is consistent with the other models in this specification. The String Event 1 bitfield (StrEvt1) exposes all of the alarms that may be found in Battery Event 1 bitfield (Evt1) in the S 802 model, in addition to some flow specific alarms such as High Pressure Alarm (HIGH_PRESSURE_ALARM). In the typical case where there is just one flow battery string this leads to some redundancy—alarms like the Over Charge Current Alarm (OVER_CHARGE_CURRENT_ALARM) will generally be set on both Evt1 in S 802 and StrEvt1 in S 807. Consumers of the communication interface should read and rely on StrEvt1 in S 807 in this case as it will always be a superset.

Module Repeating Block

A flow battery string is made up of one or more flow battery modules, and it is often necessary to monitor and control the individual modules in the string. Model S 807 includes a SunSpec repeating block which is repeated once for every module in the string. For more information on how repeating blocks are handled in the storage models, please see the **Repeating Blocks** section above.

The model size register in S 807 may be used to determine how many elements exist in the module repeating block. The Module Count (NMod) field in S 807 indicates how many of those repeating instances correspond to physical modules.

Within the model repeating block there are attributes which are similar to what is found in the fixed block of S 807. For example, where the fixed portion has a Module Count

(NMod) field which indicates the number of modules in the string, the repeating block has a Stack Count (ModNStk) field which indicates the number of stacks within the module.

Key voltage measurements for the module are provided via the Open Circuit Voltage (ModOCV) and Module External Voltage (ModV) registers. Voltage extremes are also provided through the Max Cell Voltage (ModCellVMax), Min Cell Voltage (ModCellVMin), and the associated location fields.

The Anolyte Temperature (ModAnoTmp) and Catholyte Temperature (ModCatTmp) registers provide detailed temperature information for each module in the string. Additionally, module-level alarms and warnings are exposed through the Module Event 1 (ModEvt1) and Module Event 2 (ModEvt2) attributes.

Enabling/Disabling and Connecting/Disconnecting Modules

For testing and/or maintenance purposes, it may be desirable to temporarily disconnect a module from the flow battery string. A specific module may be enabled or disabled using the Enable/Disable Module command (ModSetEna). A disabled module will not attempt to connect the next time that the string is asked to connect, and will be series bypassed before the remainder of the string connects again. An enabled module will reinstate a series connection with the remainder of the string, before the string connects again. The overall connected status of the module is indicated by the Module Status (ModSt) field. Detailed connection information for each of the contactors can be bound by reading the Contactor Status (ModConSt) field.

ModSt also indicates whether or not a given module is enabled. A flow battery module may be disabled for a variety of reasons including a fault condition, maintenance activity, or reduction in operational capacity. Additionally, modules can be disabled manually (i.e. by an operator) or automatically by the battery management system when certain conditions arise. ModDisRsn is used to convey additional context about why a particular module has been disabled. It can also be used to help distinguish modules that are potentially available for reconnection from modules that are completely unavailable.

The valid values for ModDisRsn are shown in **Table 12**.

Table 12: Module Disabled Reasons

Value	Description
NONE	The module is enabled.
FAULT	The module encountered a fault condition during operation and was automatically disabled. It may be re-enabled by the controller once the fault condition has been cleared.
MAINTENANCE	The battery is undergoing maintenance. This module cannot be enabled and should be considered locked and tagged out.
EXTERNAL	An operator or an external controller has intentionally disabled the module to remove it from operation.
OTHER	The module is disabled for some other reason.

Writing a 1 (ENABLE_MODULE) to the ModSetEna register causes the module to be enabled. Writing a 2 (DISABLE_MODULE) to the register causes the module to be disabled. When a module is disabled using ModSetEna, ModDisRsn should be set to the EXTERNAL value shown in **Table 12**.

When read, the ModSetEna register should reflect the current state in the enabling process, where 1 indicates that the module is in the process of being enabled, 2 indicates that the module is in the process of being disabled, and 0 indicates that the process is complete.

ModSetEna is optional, as the physical capability to connect or disconnect a specific module may not be implemented in a given system. If the Enable/Disable functionality is not supported, ModSetEna should return the SunSpec "Not Implemented" value for an enum16 (0xFFFF). However, it should be noted that this capability is very desirable, and all vendors are encouraged to implement ModSetEna if possible.

The Connect/Disconnect Module (ModSetCon) writeable register in the module repeating block may be used to connect or disconnect an individual module. When used to connect a module, this function must honor the enabled state of the module, as specified by ModSetEna. For example, if a module is asked to connect using ModSetCon but that string is not enabled, the operation should fail and the CONTACTOR_STATUS bit in the ModSt register should not be set.

Writing a 1 (CONNECT_MODULE) to the ModSetCon register causes the module to be connected. Writing a 2 (DISCONNECT_MODULE) to the register causes the module to be disabled.

When read, the ModSetCon register should reflect the current state in the connect/ disconnect process, where 1 indicates that the module is in the process of being connected, 2 indicates that the module is in the process of being disconnected, and 0 indicates that the process is complete.

Flow Battery Module Model (S 808)

The Flow Battery Module Model will be fully defined in the next draft of this specification (Draft 5).

Flow Battery Stack Model (S 809)

The Flow Battery Stack Model will be fully defined in the next draft of this specification (Draft 5).

Appendix A: Differences Between Draft 3 and Draft 4

Table 11 below summarizes the key differences between Draft 3 of this specification and the current version (Draft 4).

Table 12: Differences Between Draft 3 and Draft 4

Model(s) Affected	Change	Impact
802	Model 801 has been deprecated. Meaningful registers have been merged into 802.	The 801 model should no longer be implemented for any storage device, regardless of type.
802 and 803	Some S 803 registers relocated to S 802. Relocated registers include: • A • CellVMax • CellVMaxStr • CellVMaxMod • CellVMin • CellVMinStr • CellVMinMod	Now more feasible to use the 802 model in isolation (e.g. for battery types which are not specifically included in this specification).
804, 805	Added new models which provide additional detail for lithium-ion battery installations.	Manufacturers who want to provide additional detail for their batteries may optionally implement these models.
807	Added new flow battery string model.	Flow battery manufacturers may now implement a standardized communication interface.
Multiple	Event bits (e.g. Evt1 and Evt2) revised to promote consistency across models.	Consistency across event bitfields within the storage models promotes reuse and minimizes defects.
Multiple	Revised register names for consistency within the models and with regard to IEC 61850.	More consistent names ensure maximum readability.

Multiple	Added repeating block counts and repeating element indices.	Repeating blocks may now include “spares” which allow for future expansion. Index elements within detailed models such as S 804 help bridge the gap between models.