SBDS – Addendum For Fuel Cell Systems

Release 1.02
April 11, 2007

SBDS addendum for Fuel Cell System: Initial release April 11th 2007

Revision History:

Revision	Changes	Date
1.0	Initial Release	8/29/06
1.01	Correction to Auto_Soft-OFF(2Bh) function [Section 5.2.7] Release for voting	1/1907
1.02	Removed Xpress-P^x-Data(24h) as a result of review telecon held on 2/13/2007. This is because the existing ManufacturerData(23h) is able to duplicate the proposed Xpress-P^x-Data(24h) function. The command codes for those commands following 24h have all changed command codes to remove the break in sequence.	4/11/07

Table of Contents

1. Introduction

Smart Battery Data Specifications (SBDS) is an ideal solution for many of the issues related to batteries used in portable electronic equipment (such as laptop computer systems, cellular telephones or video cameras) but SBDS does not take into account Fuel Cell systems as power sources. This addendum is compatible with SBDS while adding new functions that allow greater control of Fuel Cell systems and will result in better performance. Additionally, this addendum specifically defines how Fuel Cell systems can respond to SBDS functions to remain compatible with current SBDS compatible devices.

Fuel Cell systems presently have a number of differences compared to traditional batteries. Fuel Cell systems can be turned on and off. They have a startup time during which they might not produce power or might produce only a very limited amount of power. Fuel Cell systems are refueled instead of recharged. Fuel Cell system peak power levels are generally far less than similarly sized battery packs. Fuel Cell systems degrade in a different manner than traditional batteries, becoming less efficient with age. Additionally some Fuel Cell systems may include internal rechargeable batteries.

The main differences of using Fuel Cell systems in place of batteries are: turning the Fuel Cell system "On" and the corresponding possible delay in power production, and maximum power limitations.

This addendum is not designed to limit innovation amongst battery manufacturers, but rather, provide the user and the SMBus Host with a consistent set of information about any particular Smart Battery or Fuel Cell System.

Additionally, although SMBUS and I2C interface are used as the underlying physical layer for this addendum, however these command sets can be implanted using other single or multi-wire interfaces.

1.1. Scope

This document specifies how Fuel Cell systems can return a data set that is compatible with SBDS. This document also defines new functions for Fuel Cell systems for added capabilities. This specification is generic with regard to the type of battery or Fuel Cell system chemistry, the battery or Fuel Cell system voltage, the battery pack or Fuel Cell system fuel cartridge capacity as well as the battery pack or system's physical packaging.

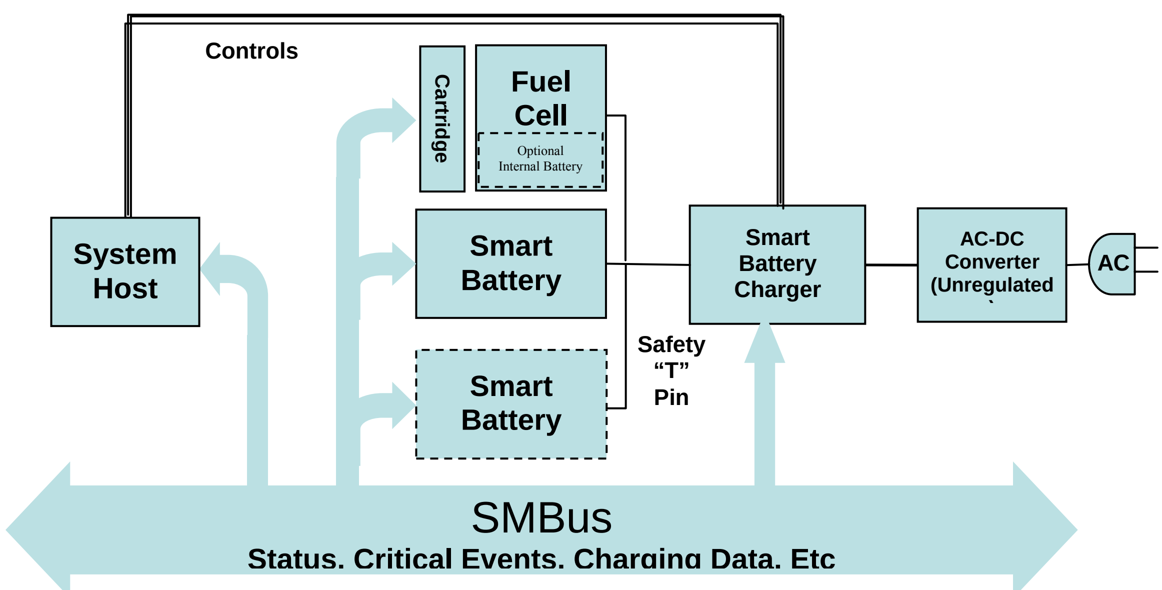
2. References

- *Smart Battery Data Specification*, Revision 1.1, SBS-Implementers Forum, December, 1998
- *Smart Battery Charger Specification*, Revision 1.1, SBS-Implementers Forum, December, 1998
- *Smart Battery Selector Specification*, Revision 1.1, SBS-Implementers Forum, December, 1998
- *Smart Battery System Manager Specification*, Revision 1.1, SBS-Implementers Forum, December, 1998
- *System Management Bus Specification*, Revision 1.1, SBS-Implementers Forum, December, 1998
- *System Management Bus BIOS Interface Specification*, Revision 1.0, February 15, 1995
- *ACPI Specifications,* Version 1.0a, Intel Corporation, Microsoft Corporation, Toshiba Corp., July 1998 (http://www.teleport.com/~acpi)
- *The I²C-bus and how to use it*, Philips Semiconductors document #98-8080-575-01.
- *ACCESS.bus Specifications -- Version 2.2*, ACCESS.bus Industry Group, 370 Altair Way Suite 215, Sunnyvale, CA 94086 Tel (408) 991-3517
- IEC SC21A - "Alkaline Secondary Cells and Batteries", IEC committee 21, Sub-committee A. (Responsible for development of standard battery pack sizes and electrical specifications.)
- IEC SC48B - "Connectors", IEC committee 48, Sub-committee B. (Responsible for development of connector standards for batteries.)

3. Definitions

- **Fuel Cell:** An electro-chemical device that converts fuel into electricity.
- **Fuel Cell System:** A system consisting of a Fuel Cell, a fuel cartridge or storage tank, possibly a battery, and hardware that controls the Fuel Cell and provides present state, calculated and predicted information to its SMBus Host under software control. The content and method are described in this specification.
- **Fuel Cartridge**: A container (either replaceable or refillable) that contains the fuel used by a Fuel Cell system to produce electricity.
- **Internal Battery**: An optional rechargeable battery inside the Fuel Cell system which can be transparent to the host which can be recharged from external sources
- **I²C-bus**: A two-wire bus developed by Phillips, used to transport data between low-speed devices.
- **Smart Battery:** A battery equipped with specialized hardware that provides present state, calculated and predicted information to its SMBus Host under software control.
- **Smart Battery Charger:** A battery charger that periodically communicates with a Smart Battery and alters its charging characteristics in response to information provided by the Smart Battery.
- **SMBus**: The System Management Bus is a specific implementation of an I²C-bus that describes data protocols, device addresses and additional electrical requirements that is designed to physically transport commands and information between the Smart Battery, SMBus Host, Smart Battery Charger and other Smart Devices.
- **SMBus Host:** A piece of portable electronic equipment powered by a Smart Battery. It is able to communicate with the Smart Battery and use information provided by the battery.

4. Fuel Cell System Overview



The Fuel Cell System communicates with other devices (such as the SMBus Host and the Smart Battery Charger) via two separate communication interfaces:

- The first uses the SMBus CLOCK and DATA lines and is the primary communication channel between the Fuel Cell System and other SMBus devices. The Fuel Cell System will provide data when requested, send charging information to the Smart Battery Charger, and broadcast critical alarm information when parameters (measured or calculated) exceed predetermined limits within the particular Fuel Cell System.

- The other required communication interface is the secondary signaling mechanism or ‘Safety Signal’ (the ‘T-pin’ on a Smart Battery pack connector). This is a variable resistance output from the Smart Battery or Fuel Cell system which indicates when charging is permitted. If this ‘Safety Signal’ pin is left open, it will signal the Smart Battery Charger that charging is not allowed.

It is possible in some cases such as a tethered fuel system to include an internal rechargeable smart battery where this battery can be recharged by an external source. In these cases the system has to be able to communicate with this battery for its status as well as its charging status and requirements over the SMBus.

4.1. Smart Battery or Fuel Cell System Software Definition

4.1.1. SMBus Host to Smart Battery or Fuel Cell System

SMBus Host to Smart Battery or Fuel Cell System communications are performed:

- To allow the user to know the Smart Battery or Fuel Cell System’s remaining runtime
- To allow Smart Batteries or Fuel Cell System’s to provide accurate information to their user
- To determine the SMBus Host's real-time power requirements
- To enable power management based on “real” values supplied by the battery or Fuel Cell system
- To enable battery or Fuel Cell system manufacturers to collect information about a Smart Battery or Fuel Cell System’s usage
- To allow battery or Fuel Cell system manufacturers to electronically "stamp" batteries or Fuel Cell systems at time of manufacture
- To allow the SMBus Host to change Fuel Cell system status (startup, shutdown, idle)

4.1.2. Smart Battery Charger to Smart Battery or Fuel Cell System (or vice versa)

Smart Battery Charger to Smart Battery or Fuel Cell System communications are performed:

- To allow Smart Batteries or Fuel Cell System internal batteries to be charged as rapidly and as safely as possible
- To allow access to the "correct" charger algorithm for the battery or Fuel Cell system internal battery.

4.1.3. Smart Battery or Fuel Cell system to SMBus Host or Smart Battery Charger

Smart Battery or Fuel Cell System to SMBus Host or Smart Battery Charger communications are performed:

- To allow the Smart Battery or Fuel Cell System to warn other system components of potential problems.
- To allow the Smart Battery or Fuel Cell System to warn the user about potentially dangerous situations that they can rectify.
- To allow the Smart Battery or Fuel Cell System internal battery to instruct the Smart Battery Charger what Charge Current and Charge Voltage it would like to be charged with.
- To allow the Fuel Cell system to indicate its status (startup, shutdown, fuel cartridge removed).

4.2. Smart Battery or Fuel Cell System Characteristics

The Smart Battery or Fuel Cell System may or may not be present in a system. Additionally, it may dynamically be added or removed while the system is powered. Therefore, it must exhibit predictable behaviors when inserted in a system and/or when the system is turned on. The following is a description of the battery or Fuel Cell system’s states and a description of the actions that take place as a result of changes.

4.2.1. Initial Conditions;

The system must detect when a Smart Battery or Fuel Cell System is present and this is done by using BatteryMode() status bit 10.as indicated by bit10 of BatteryMode() command.

Function (Data Value)	**Initial Value (Smart Battery)**	**Initial Value (Fuel Cell Systems)**	**Units**
RemainingCapacityAlarm()	10% of DesignCapacity() value	0	mAh
RemainingTimeAlarm()	10	10	minutes
BatteryMode()	Bit 15: CAPACITY_MODE=0 **Bit 14:** **CHARGER_MODE=0** **Bit 13:** **ALARM_MODE=0** **Bit 10 :** **Fuel_Cell_Mode = 0** Bit 9: PRIMARY_BATTERY=0 BIT 8: CHARGE_CONTROLLER_ENABLED=0	**Bit 15:** **CAPACITY_MODE=1** **Bit 14:** **CHARGER_MODE=1** **Bit 13:** **ALARM_MODE=1** **Bit 10 :** **Fuel_Cell_Mode** = 1 Bit 9: PRIMARY_BATTERY=0 BIT 8: CHARGE_CONTROLLER_ENABLED=0	bit value
BatteryStatus()	Bit 7: INITIALIZED=1	Bit 7: INITIALIZED=1	bit value
CycleCount()	typically less than 5	typically less than 5	decimal

4.2.2. Fuel Cell 'OFF" State

The Fuel Cell System may enter the "Off State" whenever the SMBus Clock and Data lines both remain low for greater than 2.5 seconds or driven by the host to this state. In this condition the power is removed from the Fuel Cell and no communication will occur. If the Fuel Cell system has a physical "On/Off switch", it will go into OFF state immediately when the switch is moved to the "Off" position and will stop communicating via SMBus.

Fuel Cell must enter this mode automatically in case of any critical alarms and or loss of system communications.

4.2.3. Soft-OFF State:

In this mode the Fuel Cell can communicate to the host or other system electronics. System may drive the Fuel Cell to this state from idle mode when the load no longer exists and or charging is complete.
The Fuel Cell must enter Soft OFF mode automatically after charging the optional internal battery is complete with or without the host intervention

The Fuel Cell enters this state whenever it detects that the SMBus Clock and Data lines go high and will remain in this mode until its operating mode is changed via the SFMode() function call.

The Fuel Cell may also automatically transition to Start Up state after entering Soft-Off state in order to prepare to produce power.

.The Fuel Cell system may not disrupt traffic on the SMBus, however the physical act of inserting a new device onto a live bus may cause an inadvertent communication interruption. The Smart Fuel Cell System may not begin broadcasting messages to either the SMBus Host or Smart Battery Charger for at least 10 seconds after entering the SMBus "On State." Including the Soft-Off State.

4.2.4. STARTUP State:

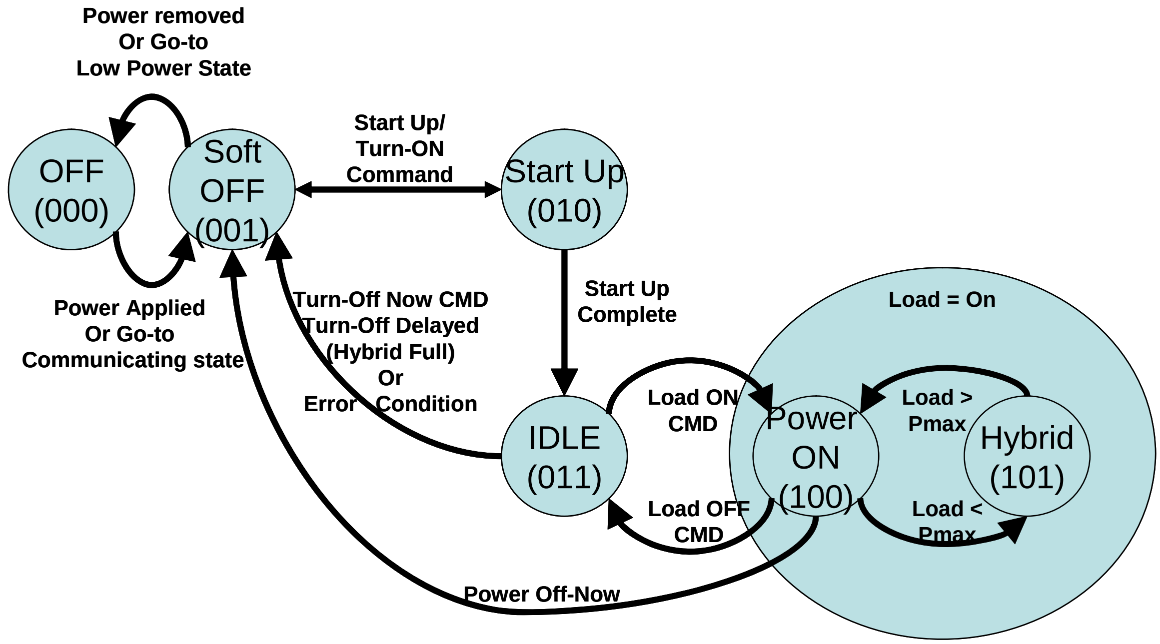
In this state, the Fuel Cell system prepares itself to provide power since the fuel cell system can not deliver the load power immediately and requires a set up time (SetupTime(0x26h). The Fuel Cell should automatically enter this state initially by changing FCMode() upon power up and when ready it should transition to Idle state awaiting command to provide power.

4.2.5. Power ON State:
In this state, the Fuel Cell system is producing power to the load and could be charging the battery at the same time.

4.2.6. Hybrid State:
Hybrid state is when the load power requirements exceed the capability of the Fuel Cell. Under this condition the Fuel Cell is working in conjunction (parallel) with battery to produce the required platform power and must be able to toggle to the Power-ON state automatically once the load's power is with in the Fuel Cell's capability.

4.2.7. Idle State
In IDLE state the Fuel Cell System could deliver power and is ready to enter the Power On state. This state could also be used as a low power state (standby) where the Fuel Cell System is awaiting for the load to be turned on.
The Fuel Cell system may also enter Idel state under alarm conditions

4.2.8 State Diagram and System Transitions



Transition Table		Host initiated	Fuel Cell initiated	Possible Conditions
OFF to Soft OFF	000 → 001	-	-	User initiated (On-Off Switch)
Soft OFF to OFF	001 → 000	√	√	Alarm, System disconnect, Turn Of
Soft OFF to Start Up	001 → 010	√	-	System ON
Start Up to Soft OFF	010 → 001	√	√	Error
Start Up to Idle	010 →011	-	√	Start-up complete, system interrupt
Idle to Soft Off	011 → 001	√	√	Error, Alarm, Turn-OFF
Idel to Power-ON/Hybrid	011 → 10x	√	-	Load ready, Toggle to battery if Po
Load Power to Idle	10x → 011	√	√	Load off command, Alarm, Error

Note: All other Transitions are invalid and ignored.

4.2.9. Safety Signal Hardware Requirements (Smart Battery Charger Interface)

The Fuel Cell System, similar to Smart Batteries, must also provide an additional signal to allow for safe charging. This 'Safety Signal' is also commonly referenced as the 'T-pin' or 'Thermistor' on some Smart Battery hardware connectors. The 'Safety Signal' is an output from the Smart Battery and may be used by a Smart Battery Charger (or other device) to determine if charging of the Smart Battery is permitted. This signal is a variable resistance output as measured between the 'Safety Signal' pin and the negative terminal of the battery.

If a Fuel Cell System has an internal rechargeable battery that has the capability of being recharged by the Smart Battery Charger, it must have a Safety Signal Resistance of 2850Ω to 31.5kΩ. Otherwise, the Safety Signal should be open (Rss > 95k Ω). If a Fuel Cell system wants to stop the Smart Battery Charger from charging its internal battery, the Safety Signal should be open (Rss > 95k Ω).

The Smart Battery Charger's capabilities are altered by the value of the Safety Signal. As a required safety feature, the charger must NOT charge a battery when it senses the resistance between the Safety Signal pin and ground to be in the range between 425 and 3150 ohms. The valid ranges of the Safety Signal are summarized below along with the charger's capabilities for the range. (Please also refer to the Smart Battery Charger Specification.)

Safety Signal Resistance, Rss Ohms (Ω)	Charger Status Bits	Description	Wake-up Charge*1	Controlled Charge*2	Notes
0 < Rss < 575	RES_UR, RES_HOT	Under-range	allowed for initial time-out period	allowed	Charger can "Wake-up Charge" for time-out period; "Controlled Charge" allowed.
425 < Rss < 3150	RES_HOT	Hot	not allowed	not allowed	Fail-safe charge termination – charger must not supply current
2850 < Rss < 31.5k	(none)	Normal range	allowed indefinitely	allowed	Charger can "Wake-up Charge" indefinitely; "Controlled Charge" allowed.
28.5k < Rss < 105k	RES_COLD	Cold	allowed for initial time-out period	allowed	Charger can "Wake-up Charge" for time-out period only.
Rss > 95k	RES_OR, RES_COLD	Over-range	not allowed	not allowed	Can be used as battery detect; charger does not supply current.

*NOTES: 1) In the table above, "Wake-up Charge" refers to a maximum amount of charge that the Smart Battery may accept prior to re-enabling itself and communicating on the SMBus. This amount of charge (maximum current and maximum time) is defined by the Smart Battery Charger Specification to be 100 mA (maximum) for 140 to 210 seconds. Removal of the Smart Battery from the Smart Battery Charger and re-insertion will reset the 'Wake-Up' charge which may then be repeated.
2) The reference to "Controlled Charge" as used in the table indicates that the Smart Battery Charger is using the Smart Battery's values of ChargingVoltage() and ChargingCurrent() to control the charging conditions. (These may be read from the Smart Battery or received from the Smart Battery via broadcasts.)

5. Smart Battery or Fuel Cell System Interface

The following functions are used by the Smart Battery or Fuel Cell System to communicate with a SMBus Host, Smart Battery Charger and other devices connected via the SMBus.

The SMBus Host, acting in the role of a SMBus **master**, uses the Read Word and Write Word protocols to communicate numeric data with the Smart Battery or Fuel Cell System. Non-numeric data, such as the ManufacturerName(), is read using the Read Block protocol.

The Host Device can obtain data that can then be either presented to a user or applied by the Device's power management system. Two types of data are available from the Smart Battery: static data and dynamic data. Static data includes items that are not changing, such as chemistry, the original capacity, or the design voltage. Dynamic data includes both measured and calculated information. Measured data is obtained by the monitoring electronics and includes items such as temperature, voltage and current. Calculated information is based on the battery or Fuel Cell system's present state and the battery or Fuel Cell system's characteristics, such as the remaining life at the present rate of drain.

The functions are described as follows:

FunctionName()   0xnn (command code)

Description: A brief description of the function.
Purpose: The purpose of the function, and an example where appropriate.
SMBus Protocol: Refer to Section 6 for details.
Required: Is this function required for SES compatible Fuel Cell Systems?
Data Type: Dynamic or Static
Input, Output or Input/Output: A description of the data supplied to or returned by the function.
The data is described as follows:

Data type:	The type of data the function conveys
Units:	The units the data is presented in
Range:	The range of valid data
Granularity:	described as percentage of a maximum value, determined by least accurate data.
Accuracy:	How "good" is the data.

Fuel Cell system: A description of how a Fuel Cell System will return information to this function to be compatible with SBDS and SES P^X.

If an optional internal smart battery exists as part of the Fuel Cell, all commands associated with this battery return the same vales as any other smart battery in the system unless noted so.

A Fuel Cell System that complies with SBDS v1.1 must support all the command codes contained in this specification. It must support the defaults as specified. Additionally, it must support all modes and functions specified except those which it can explicitly signal the presence or absence thereof (e.g. the presence of an internal charge controller and the ability to enable or disable that controller). Portions of this specification designated 'optional' are not required for compliance.

To be compatible with SBDS, Fuel Cell Systems must supply values for functions as described in the "Fuel Cell system" headings in sections 5.1.1-5.1.21 and 5.3.1-5.3.13. For extra functionality, Fuel Cell Systems must also supply values for functions as described in the "Fuel Cell system" headings in sections 5.2.1-5.2.6. To be compatible with SES function definitions, Fuel Cell Systems need only supply values for functions as described in the "Fuel Cell system" headings in sections 5.1.1-5.1.21.

5.1. Standard SBDS for Fuel Cell Systems – SES Subset

This section lists SES P^x function definitions and the values that Fuel Cell Systems will report for these functions. SES P^x functions are a subset of SBDS. For full compatibility with SBDS, Fuel Cell Systems must also return values for the functions listed in section 5.3. In a sense, this is a guide to Fuel Cell system manufacturers as to how to make Fuel Cell Systems compatible with devices designed for standard Smart Batteries.

5.1.1. BatteryMode(03h)

Description: This function reports the battery system's operational modes and capabilities, and flags minor conditions requiring attention.
Purpose: To allow the Host Device to determine the presence of Fuel Cell system and the particular data reporting formats. (See individual bit definitions).
SMBus Protocol: Read or Write Word
Required?: Yes
Data Type: Dynamic
Input/Output: unsigned int – bit mapped
Units: not applicable
Range: 0…1
Fuel Cell system: In SBDS, BatteryMode has eight reserved bits along with eight defined bits (see SBDS spec). The currently reserved bits are 2-6 and 10-12. Fuel Cell systems will use Bit,10, to report presence of Fuel Cells as an alternative source of energy The BatteryMode flag bit 10 is set when the SBDS Primary Exchange electronics are representing a fuel-cell device. This is an indication to the Host Device that there may be load limitations, start-up delays, or other aspects of Fuel Cell operation that may change operational parameters. Additionally, the Host device will recognize the new additional functions such as DesignMaxCurrent(0x25),StartTime(0x26), FuelCellRegister(0x2A) and other optional new functions defined in the SBDS addendum for Fuel Cells.

Bit 15 (read only) always is set (1) as Capacity_mode () for current is not supported for Fuel Cells (It only reports capacity in Watts)

5.1.2. Temperature(08h)

Description: Returns the battery pack's internal temperature in degrees Kelvin (ºK)
Purpose: The Temperature function provides an accurate temperature for use by the battery-powered device's thermal management system. Since Fuel Cell systems may have a variety of internal temperatures, it is more appropriate that they report ambient temperature or internal battery temperature.
SMBus Protocol: Read Word
Required?: Yes
Data Type: Dynamic
Output: unsigned int – battery temperature in tenth degree Kelvin increments
Units: 0.1 ºK
Range: 0 to +6553.5 ºK
Granularity: 1.0 ºK or better
Accuracy: ± 4 ºK
Fuel Cell system: This function is used only for reporting internal battery temperature. For reporting internal temperatures associated with Fuel Cell, FCTemp() command must be used.

5.1.3. Voltage(09h)

Description: Returns the battery pack or Fuel Cell system voltage in milli-Volts (mV).
Purpose: The Voltage function provides the battery or Fuel Cell system terminal voltage
SMBus Protocol: Read Word
Required?: Yes
Data Type: Dynamic
Output: unsigned int – battery or Fuel Cell system terminal voltage in milli-Volts
Units: mV
Range: 0 to 65,535 mV
Granularity: 0.5% of DesignVoltage
Accuracy: ± 2.0% of DesignVoltage
Fuel Cell system: returns same.

5.1.4. Current(0Ah)

Description: Returns the current being supplied through the battery or Fuel Cell system's terminals in milli-Amps (mA)
Purpose: The Current function provides a measurement of the current flowing out of the battery or Fuel Cell system.
SMBus Protocol: Read Word

Required?: Yes
Data Type: Dynamic
Output: unsigned int – discharge rate in mA increments, negative for discharge
Units: mA
Range: 0 to -32,768 mA for discharge
Granularity: 0.5% of DesignCapacity
Accuracy: ± 2.0% of DesignCapacity
Fuel Cell system: returns same**.**

5.1.5. RelativeStateOfCharge(0Dh)

Description: Returns the predicted remaining battery capacity or Fuel Cell system fuel cartridge capacity expressed as a percentage of the DesignCapacity (%)
Purpose: The RelativeStateOfCharge function is used to estimate the amount of charge remaining in the battery or fuel in the Fuel Cell system fuel cartridge(s).
SMBus Protocol: Read Word
Required?: Yes
Data Type: Dynamic
Output: unsigned int – percent of remaining capacity
Units: %
Range: 0 to 100%
Granularity: 2%
Accuracy: 10%
Fuel Cell system: Returns the predicted remaining fuel in fuel cartridge expressed as a percentage of the DesignCapacity (%). If no fuel cartridge is installed, return 0.

5.1.6. RemainingCapacity(0Fh)

Description: Returns the predicted remaining battery capacity or Fuel Cell system internal battery capacity in milli-Amp-hours (mAh)
Purpose: The RemainingCapacity function returns the battery or Fuel Cell system fuel cartridge(s) remaining capacity in absolute terms but relative to a specific discharge rate.
SMBus Protocol: Read Word
Required?: No
Data Type: Dynamic
Output: unsigned int – remaining charge in mAh
Units: mAh
Range: 0 to 65,535 mAh
Granularity: 0.5% of DesignCapacity
Accuracy: not applicable
Fuel Cell system: If the Fuel Cell system has an internal rechargeable battery, this function returns the predicted remaining capacity in that battery in milli-Amp-hours (mAh). If the Fuel Cell system does not have an internal rechargeable battery, return the result of this calculation: RelativeStateOfCharge(0Dh)* DesignCapacity(18h)

5.1.7. FullChargeCapacity(10h)

Description: Returns the predicted pack capacity or Fuel Cell system internal battery capacity when it is fully charged in milli-Amp-hours (mAh)
Purpose: The FullChargeCapacity function provide the user with a means of understanding the "tank size" of their battery or Fuel Cell system fuel cartridge(s).
SMBus Protocol: Read Word
Required?: No
Data Type: Dynamic
Output: unsigned int – estimated full charge capacity in mAh
Units: mAh
Range: 0 to 65,535 mAh
Granularity: 0.5% of DesignCapacity
Accuracy: not applicable

Fuel Cell system: If the Fuel Cell system has an internal rechargeable battery, this function returns the predicted pack capacity of that battery when it is fully charged in milli-Amp-hours (mAh). If the Fuel Cell system does not have an internal rechargeable battery, return the same value as DesignCapacity(18h)

5.1.8. AverageTimeToEmpty(12h)

Description: Returns a rolling average of the predicted remaining battery life or Fuel Cell system fuel cartridge remaining runtime in minutes.
Purpose: The AverageTimeToEmpty displays state-of-charge information in a more useful way. By averaging the estimations, the remaining time will not appear to “jump” around.
SMBus Protocol: Read Word
Required?: Yes
Data Type: Dynamic
Output: unsigned int – minutes of operation left

Units: minutes
Range: 0 to 65,535 min
Granularity: 5 minutes
Accuracy: 25 minutes
Invalid Data Indication: 65,535 indicates battery or Fuel Cell system is not being discharged.

Fuel Cell system: Returns a rolling average of the predicted remaining number of minutes of runtime of the Fuel Cell system using the currently installed fuel cartridge(s). If the Fuel Cell system is in PRODUCING POWER MODE, this number is calculated based on power out of the Fuel Cell system. If the Fuel Cell system is not in PRODUCING POWER MODE, return 65,535. If no fuel cartridge is installed, return time until Fuel Cell system enters SHUTDOWN MODE or IDLE MODE from PRODUCING POWER MODE.

5.1.9. BatteryStatus(16h)

Description: Returns the status word which contains alarm and status bit flags which indicate end-of-discharge, over-temperature, and other conditions.
Purpose: The BatteryStatus() function is used by the Host Device to get alarm and status bits, as well as error codes from the Smart Battery.
SMBus Protocol: Read Word
Required?: Yes
Data Type: Dynamic
Output: unsigned int – Status Register with alarm conditions bit mapped as follows:
*****Alarm Bits*****
0x8000 OVER_CHARGED_ALARM
0x4000 TERMINATE_CHARGE_ALARM
0x2000 reserved
0x1000 OVER_TEMP_ALARM
0x0800 TERMINATE_DISCHARGE_ALARM
0x0400 reserved
0x0200 REMAINING_CAPACITY_ALARM
0x0100 REMAINING_TIME_ALARM
*****Status Bits*****
0x0080 INITIALIZED
0x0040 DISCHARGING
0x0020 FULLY_CHARGED
0x0010 FULLY_DISCHARGED

Fuel Cell system: Returns only internal battery alarms if present. A new command FCStatus has been defined for Fuel Cell Alarm conditions.

5.1.10. ChargingCurrent(14h)

Description: Returns the Smart Battery’s desired charging rate in milli-Amps (mA)
Purpose: The ChargingCurrent function returns the maximum current that a Smart Battery Charger may deliver to the Smart Battery. In combination with the ChargingVoltage these functions permit a Smart Battery Charger to dynamically adjust its charging profile (current/voltage) for optimal charge. The battery can effectively turn off the Smart Battery

Charger by returning a value of 0 for this function. Smart Battery Chargers may be operated as a constant voltage source above their maximum regulated current range by returning a ChargingCurrent value of 65535.
Note: The Smart Battery Charger is expected to respond in one of three ways:

- Supply the current requested
- Supply its maximum current if the request is greater than its maximum and less than 65535
- Supply its maximum safe current if the request is 65535

Note: It is incumbent upon the battery or Fuel Cell system to be able to withstand considerable variations in the actual charging current supplied if the load varies rapidly during charging.
SMBus Protocol: Read Word
Required?: No
Data Type: Dynamic
Output: unsigned int – maximum charger output current in mA

Units: mA
Range: 0 to 65,535 mA
Granularity: 0.5% of DesignCapacity
Accuracy: not applicable
Invalid Data Indication: 65,535 indicates the Smart Battery Charger should operate as a voltage source outside its maximum regulated current range.

Fuel Cell system: If the Fuel Cell System has an internal rechargeable battery, this function returns that battery's desired charging rate in milli-Amps (mA). If the Fuel Cell system does not have an internal rechargeable battery, this function returns 0.

5.1.11. ChargingVoltage(15h)

Description: Returns the Smart Battery's desired charging voltage in milli-Volts (mV). This represents the maximum voltage which may be provided by the Smart Battery Charger during charging.
Purpose: The ChargingVoltage function sets the maximum voltage that a Smart Battery Charger may deliver to the Smart Battery. In combination with the ChargingCurrent function these values permit a Smart Battery Charger to dynamically adjust its charging profile (current/voltage) for optimal charge. The battery can effectively turn off the Smart Battery Charger by returning a value of 0 for this function. Smart Battery Chargers may be operated as a constant current source above their maximum regulated voltage range by returning a ChargingVoltage value of 65535.
Note: The Smart Battery Charger is expected to respond in one of three ways:

- Supply the voltage requested
- Supply its maximum voltage if the request is greater than its maximum and less than 65535
- Supply its maximum voltage if the request is 65535

SMBus Protocol: Read Word
Required?: No
Data Type: Static
Output: unsigned int –charger output voltage in mV

Units: mV
Range: 0 to 65,535 mV
Granularity: 0.5% of the DesignVoltage
Accuracy: not applicable
Invalid Data Indication: 65,535 indicates the Smart Battery Charger should operate as a current source outside its maximum regulated voltage range.

Fuel Cell system: If the Fuel Cell System has an internal rechargeable battery, this function returns that battery's desired charging voltage in milli-Volts (mV). If the Fuel Cell system does not have an internal rechargeable battery, this function returns 0.

5.1.12. CycleCount(17h)

Description: Returns the number of cycles the battery or Fuel Cell system has experienced. A cycle is defined as: An amount of discharge approximately equal to the value of DesignCapacity.
Purpose: The CycleCount function provides a means to determine the battery's wear. It may be used to give advanced warning that the battery is nearing its end of life. The CycleCount returned value multiplied by the DesignCapacity value can give an approximate "odometer" reading for the total capacity delivered by the battery.
SMBus Protocol: Read Word

Required?: No
Data Type: Dynamic
Output: unsigned int –count of charge/discharge cycles the battery or Fuel Cell system's internal battery has experienced.
Units: cycle
Range: 0 to 65,534 cycles; 65,535 indicates battery has experienced 65,535 or more cycles.
Granularity: 1 cycle
Accuracy: absolute count
Fuel Cell system: This function returns the number of start/stop cycles that Fuel Cell has accumulated over its life.

5.1.13, DesignCapacity(18h)

Description: Returns the full capacity of a new battery pack or Fuel Cell system fuel cartridge(s) in milli-Amp-hours (mAh). This value is expressed in mAh at a C/5 discharge rate.
Purpose: The DesignCapacity function provides a static value representing the full charge capacity of the battery pack or Fuel Cell system fuel cartridge(s). It can be used along with the RelativeStateOfCharge value to determine a remaining capacity value in mAh.
SMBus Protocol: Read Word
Required?: Yes
Data Type: Dynamic
Output: unsigned int – the battery or Fuel Cell system fuel cartridge capacity in mAh
Units: mAh
Range: 0 to 65,535 mAh
Fuel Cell system: Returns the full capacity in milli-Amp-hours (mAh) of the fuel cartridge(s) being used by the Fuel Cell system. This value is expressed in mAh and is calculated using the Fuel Cell system's rated efficiency at a rolling average of its current power usage. If no fuel cartridges are installed, return 0.

5.1.14. DesignVoltage(19h)

Description: Returns the design voltage of a new battery pack or Fuel Cell system in milli-Volts (mV).
Purpose: The DesignVoltage function can be used to give additional information about a particular battery or Fuel Cell system's terminal voltage.
SMBus Protocol: Read Word
Required?: Yes
Data Type: Static
Output: unsigned int –the battery or Fuel Cell system's designed terminal voltage in mV
Units: mV
Range: 0 to 65,535 mV
Fuel Cell system: returns same.

5.1.15. ManufactureDate(1Bh)

Description: This function returns the date the cell pack was manufactured in a packed integer. The date is packed in the following fashion: (Year-1980)*512+Month*32+day.
Purpose: The ManufactureDate provides the system with information that can be used to uniquely identify a particular battery or Fuel Cell system. It also helps indicate the age of the battery.
SMBus Protocol: Read Word
Required?: Yes
Data Type: Static
Output: unsigned int – packed date of manufacture

Field	Bits Used	Format	Allowable Values
Day	0...4	5 bit binary value	1 – 31 (corresponds to date)
Month	5...8	4 bit binary value	1 – 12 (corresponds to month number)
Year	9...15	7 bit binary value	0 – 127 (corresponds to year offset by 1980)

Fuel Cell system: returns same.

5.1.16. SerialNumber(1Ch)

Description: This function is used to return a serial number. This number when combined with the ManufacturerName, the DeviceName, and the ManufacturerDate will uniquely identify the battery.
Purpose: The SerialNumber function is used to help identify a particular battery or Fuel Cell system.
SMBus Protocol: Read Word
Required?: Yes
Data Type: Static
Output: unsigned int
Range: 0 to 65,535
Fuel Cell system: returns same.

5.1.17. ManufacturerName(20h)

Desciption: This function returns a character array containing the battery or Fuel Cell manufacturer's name. For example, "BestBatt" would identify the battery or Fuel Cell's manufacturer as BestBatt.
Purpose: The ManufactureName function returns the name of the battery or Fuel Cell system manufacturer or assembler and is used for identification.
SMBus Protocol: Read Block
Required?: Yes
Data Type: Static
Output: string – character string – LIMITED TO EIGHT (8) CHARACTERS
Fuel Cell system: returns same.

5.1.18. DeviceName(21h)

Description: This function returns a character string that contains the battery or Fuel Cell's name. For example, a DeviceName of "SamrtB" would indicate that the battery is a model SmartB.
Purpose: The DeviceName function returns the battery or Fuel Cell system's name and is used to uniquely identify the battery or Fuel Cell system.
SMBus Protocol: Read Block
Required?: Yes
Data Type: Static
Output: string – character string – LIMITED TO EIGHT (8) CHARACTERS
Fuel Cell system: returns same.

5.1.19. DeviceChemistry(22h)

Description: This function returns a character string that contains the battery or Fuel Cell's chemistry. For example, if the DeviceChemistry function returns "LS02," the battery pack would contain primary lithium cells.
Purpose: The DeviceChemistry function gives cell chemistry information.
SMBus Protocol: Read Block
Required?: Yes
Data Type: Static
Output: string – character string – LIMITED TO FOUR (4) CHARACTERS

Note: The following is a partial list of chemistries and their expected abbreviations. These abbreviations are NOT case sensitive.

Lithium Sulfur Dioxide	LSO2
Lithium Manganese Dioxide	LMnO
Lithium	LCFx
Lead Acid	PbAc
Lithium Ion	LION
Nickel Cadmium	NiCd
Nickel Metal Hydride	NiMH
Nickel Zinc	NiZn
Rechargeabe Alkaline-Manganese	RAM
Zinc Air	ZnAr
Lithium Polymer	LiP

Additional chemistry types:

Hydrogen Fuel Cell	**H2FC**
NaBH Fuel Cell	**BHFC**
Reformed Methanol Fuel Cell	**RMFC**
Direct Methanol Fuel Cell	**DMFC**
Formic Acid Fuel Cell	**FAFC**
Butane Fuel Cell	**BSFC**
Propane Fuel Cell	**PSFC**
Solid Oxide Fuel Cell	**SOFC**

5.1.20. ManufacturerData(23h) – OPTIONAL

Description: This function is optional and its meaning is implementation specific. It may be used by a battery manufacturer or silicon supplier to return specific version information, internal calibration information, or some other manufacturer specific function. There is no implied or required use for this function and therefore it may be used for multiple purposes. Fuel Cell systems may use this to return additional system information including internal temperatures, component status, and additional error codes.
Purpose: This function's purpose is manufacturer specific. No functional requirement is implied although example uses are mentioned in this text.
SMBus Protocol: Read Block (Write Block Optional)
Required?: No
Data Type: Dynamic
Output: Block – Content determined by the Smart Battery's manufacturer
Fuel Cell system: returns same.

5.2. SBDS Addendum – Additional Functions for Fuel Cell Systems

For compatibility with Fuel Cell systems, several new required functions need to be added to the standard set of SBDS defined functions

5.2.1. DesignMaxPower(24h)

Description: Returns the maximum continuous net current of a Fuel Cell system in milli-Watts (mW).
Purpose: The DesignMaxCurrent function provides the maximum continuous power that a Fuel Cell system can provide. **SMBus Protocol**: Read Word
Required?: Yes
Data Type: Dynamic
Output: unsigned int – design maximum current in milli watt increments
Units: 10 milli Watts
Range: 0 to 65,535

5.2.2. StartTime(25h)

Description: Returns the time until the Fuel Cell System is capable of providing its DesignMaxPower in seconds (s)
Purpose: The StartTime is useful when hybridizing with a second power source to determine the most efficient usage of the Fuel Cell system. If the Fuel Cell system is in Power ON mode StartTime returns 0. If the Fuel Cell system is in STARTUP MODE or IDLE MODE, StartTime returns a value calculated from real internal and environmental conditions. If the Fuel Cell system is in SHUTDOWN MODE StartTime returns either a standard startup time for the Fuel Cell system, or a value calculated from real time internal and environmental conditions. If the Fuel Cell System can provide instant power out, return 0. If the Fuel Cell System does not know when it will be able to provide power, return 65,535
SMBus Protocol: Read Word
Required?: Yes
Data Type: Dynamic
Output: unsigned int – start time in s
Units: s
Range: 0 to 65,534 s. 65,535 means system can't predict time.

5.2.3. TotalRuntime(26h)

Description: Returns the total runtime that the battery or Fuel Cell system has accumulated over its life.
Purpose: The TotalRuntime function provides a means to determine the Fuel Cell system's wear. It may be used to give advanced warning that the Fuel Cell system is nearing its end of life. Although serving a similar purpose as CycleCount, TotalRuntime is a better indicator of system wear for some Fuel Cell systems. If the battery or Fuel Cell system can't calculate its total runtime, it may return a value of: CycleCount*DesignCapacity for this function.
SMBus Protocol: Read Word
Required?: No
Data Type: Dynamic
Output: unsigned int – count of runtime in hours that the battery or Fuel Cell system has experienced.
Units: hours
Range: 0 to 65,534 hours; 65,535 indicates system has experienced 65,535 or more hours of total runtime.
Granularity: 1 hour
Accuracy: 5 hours

5.2.4. FCTemp(27h)

Description: Returns the internal temperature of Fuel Cell stack and or any reformers associated with the Fuel Cell in a two byte format. The upper byte (MSB) reports the Fuel Cell's STACK temperature while the lower byte (LSB) reports the Reformer's temperature
SMBus Protocol: Read Word
Required?: Yes
Data Type: Dynamic
Output: unsigned int
Units: Temperature in degrees Celsius.
Granularity: 1 or 2 degrees Celsius based on FCMode() bit 12 setting
Range: 0 – 255 / 511 Degrees Celsius

5.2.5. FCStatus(28h)

Description: This function communicates various Fuel Cell status, presence of fuel cartridge and internal battery .
Purpose: to allow communication between Fuel Cell and system electronics
SMBUS protocol, Read
Required: Yes
DataType: Dynamic
Input / Output:
Unsigned integer bit mapped
Units: N/A

15	MSB						8	7	LSB						0
R	R	res	res	R	R	R	R	res	res	res	res	res	R	R	R

The following table summarizes the meanings of the individual bits of the FCStatus() word and specifies the default values if any..

Field	Bits Used	Format	Allowable Value
Internal_battery	15	Read	0= absence of optional internal battery 1= presence of optional internal battery
Cartridge_Status	14	Read	0= absence of fuel cartridge 1= presence of fuel cartridge
Reserved	13-12	-	
Alarms conditions	8-11	Read	1111 = INITIALIZED 1000 = OVERLOAD 0111 = CELL DAMAGE (MEA) 0110 =OVER_TEMP_ALARM 0101 = REMAINING FUEL_CAPACITY_ALARM 0100 = REMAINING_FUEL_TIME_ALARM 0011 =INSUFFICIENT_AIR_FLOW 0010 = LIQUID LEAKAGE 0001 = BOP DAMAGE (PUMP/COMPRESSOR/ETC) 0000 = NO ALARMS All other values reserved for future use
Reserved	3-7	-	
State	2-0	Read	000 Fuel Cell is in OFF state 001 Fuel Cell is in soft-OFF state (responds to commands) 010 = Fuel Cell is in Startup mode 011 = Fuel Cell is in Idle mode (load off) 100 = Fuel Cell is ON and producing power or charging 101 = Fuel Cell is ON and producing power in parallel with battery (running in Hybrid mode) Other values: Reserved

Specific Definitions of each bit:

Internal_Battery: Setting this bit indicates the existence of an internal battery used only by Fuel Cell system for startup.
Cartridge_Status: This bit indicates the presence or absence of fuel cartridge

Alarm condtions_: These bits indicate alarm conditions which require host or controller service. For example if INSUFFICIENT AIR FLOW alarm is set whenever the Fuel Cell system detects that it is not able to breathe enough air to produce full power. When this condition is set, load should be reduced or shutoff.
This alarm bits must be cleared when the Fuel Cell system detects that the condition no longer exists.

State: refer to Fuel Cell systems modes for description of these modes

5.2.6 FCMode (29h)

Description: This function configures various Fuel Cell states and or settings.
Purpose: to allow changing Fuel Cell mode of operation by the Fuel Cell or Host
SMBUS protocol, Read or Write Word
Required: Yes
DataType: Dynamic
Input / Output:
Unsigned integer bit mapped
Units: N/A.

15	MSB						8	7	LSB						0
res	res	RW	RW	res	res	res	res	res	res	res	res	R/W	R/W	R/W	RW

The following table summarizes the meanings of the individual bits of the FCMode() word and specifies the default values if any

Field	Bits Used	Format	Allowable Value
Reserved	14-15	-	
Fuel Cell Selector	13	R/W	0= information from or to internal battery 1=Information from or to Fuel Cell
FCTemp_granularity	12	R/W	0 = 1 degrees C (FCTemperature()) 1 = 2 degrees C (FCTemperature())
Reserved	11-4	-	
Change Status Enable	3	R/W	0 = FCMode() change disabled 1 = FC(mode() change enabled
Set-Mode	0-2	R/W	000 Set Fuel Cell to off State 001 Set Fuel Cell to soft-off sate (responds to commands) 010 = Set Fuel Cell to startup mode 011 = Set Fuel Cell to idle mode (load off) 100 = Set Fuel Cell to ON and producing power or charging 101 = Set Fuel Cell to ON and producing power in parallel with battery (running in Hybrid mode) Other values: Reserved

Specific Definitions for each bit:
Fuel Cell Selector: This bit selects the targeted component by the host for communication which could be either the Fuel Cell or its internal battery
FCTemp_granularity: This bit sets the maximum possible temperature ranges. See FCTemp()

Change Status Enable: Setting this bit allows changes via FCMode() command.
Set-Mode: Same as State mode in FCStatus() command, except that by using FCMode() command one can change the operating state of the Fuel Cell.

5.2.7 Auto_Soft-OFF(2Ah)

Description: This command transitions the Fuel Cell to the Soft OFF
Purpose: Auto-Soft-OFF is useful when system has detected charge completion and or load removal, however is awaiting possible changes in system usage whereby it allows adoption of power management policies.
SMBus Protocol: Read Word
Required?: Optional
Data Type: Dynamic
Output: unsigned int – awaiting Soft-Off command
Units: s
Range: 0 to 65,535 s.

5.3. SBDS for Fuel Cell Systems – In Addition to the SES Subset

This section lists standard SBDS function definitions not already described This section, along with section 5.1 is a guide to Fuel Cell system manufacturers as to how to make Fuel Cell Systems compatible with devices designed for standard Smart Batteries.

5.3.1. ManufacturerAccess() (0x00)

Description: This function is optional and its meaning is implementation specific.
Purpose: The ManufacturerAccess() function's purpose is manufacturer specific.
SMBus Protocol: Read or Write Word
Data Type: Static
Input/Output: word -- Content determined by the Smart Battery's manufacturer
Fuel Cell system: Returns same.

5.3.2. RemainingCapacityAlarm() (0x01)

Description: Sets or gets the Low Capacity alarm threshold value. Whenever the RemainingCapacity() falls below the Low Capacity value, the Smart Battery or Fuel Cell System sends AlarmWarning() messages to the SMBus Host with the REMAINING_CAPACITY_ALARM bit set. A Low Capacity value of 0 disables this alarm.
Purpose: The RemainingCapacityAlarm() function can be used by systems to indicate a first level near end of discharge state.
SMBus Protocol: Read or Write Word
Data Type: Dynamic
Input/Output: unsigned int -- value below which Low Capacity messages will be sent
Units: mAh @ C/5 or DesignMaxCurrent() value
Range: 0 to 65,534 mAh
Accuracy: see RemainingCapacity()
Fuel Cell system: If the Fuel Cell System has an internal rechargeable battery and is in STARTUP MODE or IDLE MODE, return the value needed for the internal battery to startup or finish starting up the system. If no fuel cartridges are installed, or if the Fuel Cell System is in PRODUCING POWER MODE or SHUTDOWN MODE, return 0.

5.3.3. RemainingTimeAlarm() (0x02)

Description: Sets or gets the Remaining Time alarm value. Whenever the AverageTimeToEmpty() falls below the Remaining Time value, the Smart Battery sends AlarmWarning() messages to the SMBus Host with the REMAINING_TIME_ALARM bit set. A Remaining Time value of 0 effectively disables this alarm.
Purpose: The RemainingTimeAlarm() function can be used by systems that want to adjust when the remaining time alarm warning is sent.
SMBus Protocol: Read or Write Word
Data Type: Dynamic
Input/Output: unsigned int -- the point below which Remaining Time messages will be sent

Units: minutes
Range: 0 to 65,535 minutes
Accuracy: see AverageTimeToEmpty()

Fuel Cell system: Returns same. Default is 10 minutes.

5.3.4. AtRate(04h)

Description: The AtRate() function is the first half of a two-function call-set used to set the AtRate value used in calculations made by the AtRateTimeToFull(), AtRateTimeToEmpty(), and AtRateOK() functions.

Purpose:

- When the AtRate value is positive, the AtRateTimeToFull() function returns the predicted time to fullcharge at the AtRate value of charge.
- When the AtRate value is negative, the AtRateTimeToEmpty() function returns the predicted operating time at the AtRate value of discharge.
- When the AtRate value is negative, the AtRateOK() function returns a Boolean value that predicts the battery or Fuel Cell system's ability to supply the AtRate value of ***additional*** discharge energy for a minimum of 10 seconds.

SMBus Protocol: Read or Write Word
Data Type: Dynamic
Input/Output: unsigned int -- the point below which Remaining Time messages will be sent

Units: mA
Charge Range: 1 to 32,767 mA
Discharge Range: -1 to -32,768 mA
Granularity: 1 unit

Fuel Cell system: See AtRateTimeToFull(), AtRateTimeToEmpty(), and AtRateOK() functions

5.3.5. AtRateTimeToFull(05h)

Description: Returns the predicted remaining time to fully charge the battery or Fuel Cell system internal battery at the previously written AtRate value in mA.
Purpose: The AtRateTimeToFull() function is part of a two-function call-set used to determine the predicted remaining charge time at the AtRate value (mA). Refer to AtRate() for additional usage information.
SMBus Protocol: Read Word
Data Type: Dynamic
Output: unsigned int -- predicted time in minutes to fully charge the battery

Units: minutes
Range: 0 to 65,534 min
Granularity: 2 min or better
Accuracy: ±MaxError()∗FullChargeCapacity() ÷ AtRate()
Invalid Data Indication: 65,535 indicates the battery is not being charged

Fuel Cell system: Returns same for Fuel Cell system internal battery. If the Fuel Cell System does not have an internal rechargeable battery, this function returns 65,535.

5.3.6. AtRateTimeToEmpty(06h)

Description: Returns the predicted remaining operating time if the battery or Fuel Cell system fuel cartridge is discharged at the previously written AtRate value.
Purpose: The AtRateTimeToEmpty() function is part of a two-function call-set used to determine the remaining operating time at the AtRate value. Refer to AtRate() for additional usage information.
SMBus Protocol: Read Word
Data Type: Dynamic
Output: unsigned int -- estimated operating time left

Units: minutes
Range: 0 to 65,534 min
Granularity: 2 min or better

Accuracy: −0,+MaxError()∗DesignCapacity() ÷ AtRate()
Invalid Data Indication: 65,535 indicates the battery or Fuel Cell system is not being discharged

Fuel Cell system: Returns the predicted remaining time until the attached fuel cartridge(s) are empty if the Fuel Cell system fuel cartridge is discharged at the previously written AtRate value.

5.3.7. AtRateOK(07h)

Description: Returns a Boolean value that indicates whether or not the battery or Fuel Cell system can deliver the previously written AtRate value of **additional energy** for 10 seconds (Boolean). If the AtRate value is zero or positive, the AtRateOK() function will ALWAYS return true.
Purpose: The AtRateOK() function is part of a two-function call-set used by power management systems to determine if the battery or Fuel Cell system can safely supply enough energy for an additional load. Refer to AtRate() for additional usage information.
SMBus Protocol: Read Word
Data Type: Dynamic
Output: Boolean -- indicates if the battery can supply the **additional** energy requested

Units:	Boolean
Range:	TRUE (non-zero), FALSE (zero)
Granularity:	not applicable
Accuracy:	not applicable

Fuel Cell system: Returns same. If the AtRate value is positive, return TRUE. If the AtRate value is negative, and if – AtRate() value – Current() < DesignMaxCurrent(), then return TRUE, otherwise return FALSE.

5.3.8. AverageCurrent(0Bh)

Description: Returns a one-minute rolling average based on the current being supplied (or accepted) through the battery or Fuel Cell system's terminals (mA).
Purpose: The AverageCurrent() function provides the average current flowing into or out of the battery or Fuel Cell system for the power management system.
SMBus Protocol: Read Word
Data Type: Dynamic
Output: signed int -- charge/discharge rate in mA increments - positive for charge, negative for discharge

Units:	mA
Range:	0 to 32,767 mA for charge or 0 to -32,768 mA for discharge
Granularity:	0.2% of the DesignCapacity() or better
Accuracy:	±1.0% of the DesignCapacity()

Fuel Cell system: Returns same.

5.3.9. MaxError(0Ch)

Description: Returns the expected margin of error (%) in the state of charge calculation. For example, when MaxError() returns 10% and RelativeStateOfCharge() returns 50%, the Relative StateOfCharge() is actually between 50 and 60%.
Purpose: The MaxError() function does not exist on most systems today.
SMBus Protocol: Read Word
Data Type: Static
Output: unsigned int -- percent uncertainty for selected information

Units:	%
Range:	0 to 100%
Granularity:	1%
Accuracy:	not applicable

Fuel Cell system: Returns uncertainty of fuel cartridge fill estimate. Default value for new cartridge is 5%. As cartridge empties, typical value may increase to 10% or more. If no fuel cartridge is installed, return 10%

5.3.10. AbsoluteStateOfCharge(0Eh)

Description: Returns the predicted remaining battery or Fuel Cell system fuel cartridge capacity expressed as a percentage of DesignCapacity() (%).

Purpose: See RelativeStateOfCharge() function description.
SMBus Protocol: Read Word
Data Type: Dynamic
Output: unsigned int -- percent of remaining capacity

Units:	%
Range:	0 to 100+%
Granularity:	1%
Accuracy:	-0, +MaxError()

Fuel Cell system: Returns the predicted remaining fuel cartridge capacity expressed as a percentage of DesignCapacity() (%). If no fuel cartridge is installed, return 0.

5.3.11. RunTimeToEmpty(11h)

Description: Returns the predicted remaining battery life or Fuel Cell system fuel cartridge runtime at the present rate of discharge (minutes).
Purpose: The RunTimeToEmpty() can be used by the power management system to get information about the relative gain or loss in remaining life in response to a change in power policy. This information is NOT the same as the AverageTimeToEmpty(), which is not suitable to determine the effects that result from a change in power policy.
SMBus Protocol: Read Word
Data Type: Dynamic
Output: unsigned int -- minutes of operation left

Units:	minutes
Range:	0 to 65,534 min
Granularity:	2 min or better
Accuracy:	-0, +MaxError() * DesignCapacity() / Current()

Invalid Data Indication: 65,535 indicates battery is not being discharged

Fuel Cell system: Returns the predicted remaining fuel cartridge runtime in minutes at the present rate of discharge. If no fuel cartridge is installed, return 0.

5.3.12. AverageTimeToFull(13h)

Description: Returns a one minute rolling average of the predicted remaining time until the Smart Battery or Fuel Cell System's internal battery reaches full charge (minutes).
Purpose: The AverageTimeToFull() function can be used by the SMBus Host's power management system to aid in its policy. It may also be used to find out how long the system must be left on to achieve full charge.
SMBus Protocol: Read Word
Data Type: Dynamic
Output: unsigned int -- remaining time in minutes

Units:	minutes
Range:	0 to 65,534 min
Granularity:	2 min or better
Accuracy:	±MaxError() * FullChargeCapacity() / AverageCurrent()

Invalid Data Indication: 65,535 indicates the battery is not being charged

Fuel Cell system: If the Fuel Cell system has an internal rechargeable battery, this function returns a one minute rolling average of the predicted remaining time until it reaches full charge (minutes). If the Fuel Cell system does not have an internal rechargeable battery, return 65,534.

5.3.13. SpecificationInfo(1Ah)

Description: Returns the version number of the Smart Battery specification the battery pack or Fuel Cell system supports, as well as voltage and current **and capacity** scaling information in a packed unsigned integer. Power scaling is the product of the voltage scaling times the current scaling.
This value may also indicate a version of SMBus error checking implementation. Refer to the SMBus Specification for actual implementation information.
The SpecificationInfo is packed in the following fashion: (major version number * 0x10 + minor revision number) + (voltage scaling + current scaling * 0x10) * 0x100.

Purpose: The SpecificationInfo() function is used by the SMBus Host's power management system to determine what information the Smart Battery can provide. It can be used by Smart Battery Systems where the defined 16-bit data values do not provide enough range for higher power applications.
SMBus Protocol: Read Word
Data Type: Static
Output: unsigned int -- packed specification number and scaling information

Field	Bits Used	Format	Allowable Values
Revision	0...3	4 bit binary value	0001 - Version 1.0 and 1.1 all other values reserved
Version	4...7	4 bit binary value	0001 – Version 1.0 0010 – Version 1.1 0011 - Version 1.1 with optional PEC support all other values reserved
VScale	8...11	4 bit binary value	0 - 3 (multiplies voltages* by 10 ^ VScale)
IPScale	12...15	4 bit binary value	0 - 3 (multiplies currents* and capacities by 10 ^ IPScale)

Note: Except ChargingVoltage() and ChargingCurrent() values. Example: The specification version supported by a particular battery is 1.0 and all current readings are to be scaled by a factor of 10. Power readings will be scaled by the voltage factor times the current factor (10^0 10 ^ 1) or 10 in this case. SpecificationInfo() will return 4112 (0x1010).
Fuel Cell system: Returns same.

Appendix A. Required functions for Fuel Cell Systems

The table below shows an overview of the functions included in SES P^X and standard SBDS.

Blue	Definitions used in SES PX and standard SBDS
Yellow	Additional definitions used in SES PX, and standard SBDS
Grey	Additional definitions used in standard SBDS
Red	New definitions needed for Fuel Cells

	Functions used in standard SBDS (standard definitions for smart batteries)	Functions needed for Fuel Cells (with internal rechargeable battery)	Functions needed for Fuel Cells (without internal rechargeable battery)
1	ManufacturerAccess	Same	Same
2	RemainingCapacityAlarm	Same	Same
3	RemainingTimeAlarm	Same	Same
4	BatteryMode	BatteryMode (Bit 10)	BatteryMode(Bit 10)
5	AtRate	Same	Same
6	AtRateTimeToFull	Same	N/A
7	AtRateTimeToEmpty	Same	N/A
8	AtRateOK	Same	Same
9	Temperature	Internal battery only	N/A (new temp call)
10	Voltage	Same	Same
11	Current	Same	Same
12	AverageCurrent	Same	Same
13	MaxError	Same	Same
14	RelativeStateOfCharge	Same	N/A
15	AbsoluteStateOfCharge	Same	N/A
16	RemainingCapacity	Same	Same
17	FullChargeCapacity	Same	Same
18	RunTimeToEmpty	Same	Same
19	AverageTimeToEmpty	Same	Same
20	AverageTimeToFull	Same	Same
21	ChargingCurrent	Same	N/A
22	ChargingVoltage	Same	N/A
23	BatteryStatus	Same	Same
24	CycleCount	Same	Same
25	DesignCapacity	Same	Same
26	DesignVoltage	Same	Same
27	SpecificationInfo	Same	Same
28	ManufactureDate	Same	Same
29	SerialNumber	Same	Same
30	ManufacturerName	Same	Same
31	DeviceName	Same	Same
32	DeviceChemistry	Same+ Additional Types	Same + Additional Types
33	ManufacturerData	Same	Same
35	DesignMaxPower	New Call	New Call
36	StartTime	New Call	New Call
37	Total RunTime	New Call	New Call
38	FCtemp	New Call	New Call
39	FCStatus	New Call	New Call
40	FCMode	New Call	New Call
41	Aut-Soft-OFF	New Call	New Call
42	OptionalMfgFunction5		
43	OptionalMfgFunction4		

44	OptionalMfgFunction3		
45	OptionalMfgFunction2		
46	OptionalMfgFunction1		

Appendix B. The command set in tabular form

The following table summarizes the Smart Battery or Fuel Cell System command set. It includes the function name, code, access (r,w), and data type. For a battery or Fuel Cell system to be recognized as a Smart Battery or Fuel Cell System, it must support all the functions described by this specification.

Slave Functions	Code	Access	Data
ManufacturerAccess	0x00	r/w	word
RemainingCapacityAlarm	0x01	r/w	mAh
RemainingTimeAlarm	0x02	r/w	minutes
BatteryMode	0x03	r/w	bit flags
AtRate	0x04	r/w	mA
AtRateTimeToFull	0x05	r	minutes
AtRateTimeToEmpty	0x06	r	minutes
AtRateOK	0x07	r	Boolean
Temperature	0x08	r	0.1°K
Voltage	0x09	r	mV
Current	0x0a	r	mA
AverageCurrent	0x0b	r	mA
MaxError	0x0c	r	percent
RelativeStateOfCharge	0x0d	r	percent
AbsoluteStateOfCharge	0x0e	r	percent
RemainingCapacity	0x0f	r	mAh
FullChargeCapacity	0x10	r	mAh
RunTimeToEmpty	0x11	r	minutes
AverageTimeToEmpty	0x12	r	minutes
AverageTimeToFull	0x13	r	minutes
ChargingCurrent	0x14	r	mA
ChargingVoltage	0x15	r	mV
BatteryStatus	0x16	r	bit flags
CycleCount	0x17	r	count
DesignCapacity	0x18	r	mAh
DesignVoltage	0x19	r	mV
SpecificationInfo	0x1a	r	unsigned int
ManufactureDate	0x1b	r	unsigned int
SerialNumber	0x1c	r	number
reserved	0x1d - 0x1f		
ManufacturerName	0x20	r	string
DeviceName	0x21	r	string
DeviceChemistry	0x22	r	string
ManufacturerData	0x23	r	data
DesignMaxPower	0x24	r	data
StartTime	0x25	r	data
TotalRuntime	0x26	r	data
FCTemp	0x27	r	data
FCStatus	0x28	r	bit flags
FCMode	0x29	r/w	bit flags
Auto-Soft_OFF	0x2a	r	data
reserved	0x2b-0x2e		
OptionalMfgFunction5	0x2f	r/w	data

reserved	0x30-0x3b		
OptionalMfgFunction4 OptionalMfgFunction3-1	0x3c 0x3d- 0x3f	r/w r/w	word word