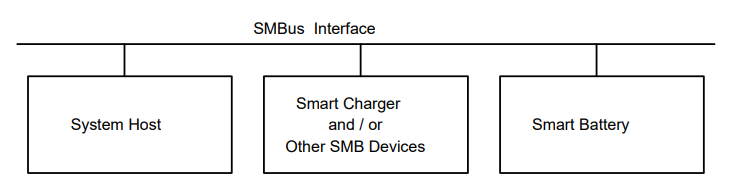
**1 Введение.**

Smart battery specification представляет идеальное решение для многих случаев, связанных с батареями, используемыми в портативной электронике, такой как laptop computer systems, сотовые телефоны или видеокамеры. Батареи в настоящее время имеют ряд ограничений как с точки зрения пользователя, так и с точки зрения аппаратуры. Прежде всего, они представляют собой непредсказуемый источник энергии. Как правило, пользователь не знает заранее, что батарея скоро может сесть, или как долго их батарея еще сможет работать. Во-вторых, оборудование, работающее от батареи, не может определить, может ли аккумулятор в его нынешнем состоянии обеспечить достаточную мощность для дополнительной нагрузки (например, раскрутка жесткого диска). В-третьих, зарядные устройства для аккумулятора должны быть индивидуально адаптированы для использования с конкретным типом химического элемента и могут привести к повреждению элемента, если они для него не предназначены.

Эта спецификация, как будет показано ниже, определяет потоки данных, проходящие по SMBus между Smart Battery, SMBus хостом, зарядным устройством и другими устройствами. Более детальное описание электрического интерфейса и протоколов данных может быть найдено в дополнительной документации (обратитесь в раздел ссылок).



Основные компоненты SMBus интерфейса:

**Электрика**: обратитесь к System Management Bus Specification для большей информации.

**Протокол**: обратитесь к System Management Bus Specification для большей информации.

**Данные**: описаны в данной спецификации.

Данная спецификация определяет информацию, которую Smart Battery поставляет пользователю. Он не предназначен для ограничения инноваций среди производителей аккумуляторов, а скорее обеспечивает пользователю и SMBus-хосту согласованный набор информации о какой-либо конкретной батарее.

**3 Терминология.**

**ACPI**: Advanced Configuration and Power Interface. Определение интерфейса операционной системы для управления электропитанием и управляющих функций. Это позволяет большему количеству ОС контролировать аппаратуру системы и позволяет осуществлять расширенное управление питанием.

**APM**: Advanced Power Management. Интерфейс BIOSпредназначенный для разрешения системного управления питанием через программное обеспечение.

**Battery**: одна или более ячеек, которые предназначены для выдачи электрической энергии.

**Cell**: элементарная ячейка в батарее. Большинство батарей состоят из нескольких ячеек, соединенных последовательно, параллельно, или последовательно-параллельно.

**I2C шина**: двухпроводная шина, разработанная Phillips, используемая для пересылки данных между низкоскоростными устройствами.

**Smart Battery**: батарея, снабженная специализированной аппаратурой, которая предоставляет текущее состояние, рассчитываемую и прогнозируемую информацию своему SMBus хост-контроллеру и находящаяся под программным управлением. Содержание и метод описаны в этой спецификации.

**Smart Battery Charger**: зарядное устройство, которое периодически взаимодействует с Smart Battery и управляет процессом заряда в зависимости от информации, получаемой от Smart Battery.

**Packet Error Check (PEC)**: дополнительный байт протокола SMBus, который используется для проверки ошибок при передаче в SMBus. Обратитесь к System Management Bus Specification Revision 1.1. Smart Battery индицирует возможность поддержки этого пакета с помощью значения version в функции SpecificationInfo().

**4.1 Модель Smart Battery.**

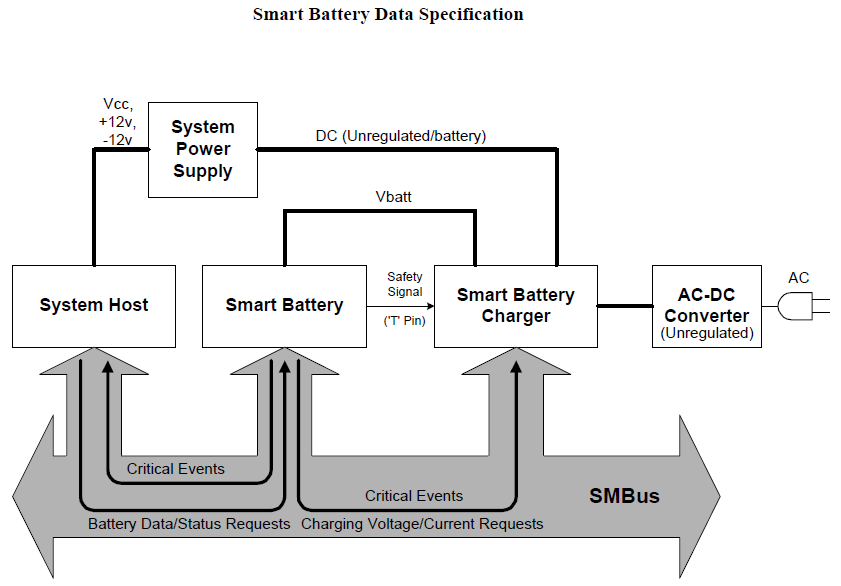
Одна из возможных моделей Smart Battery представляет собой систему, состоящую из батареи, зарядного устройства и хоста (ноутбук, компьютер, видеокамера, сотовый телефон или другое портативное оборудование). Так как это система, важно рассмотреть компоненты и их взаимодействие. (Системы могут включать дополнительные компоненты, такие как дополнительные батареи, селекторы батарей, температурные датчики, и/или другие SMBus устройства).

Smart Battery состоит из набора ячеек, способных выдавать энергию. Электроника, включенная в состав Smart Battery может следить за отдельными параметрами среды для того, чтобы вычислить соответствующие значения данных, требуемых спецификацией. Электроника не обязательно должна быть размещена внутри Smart Battery, если батарея не извлекаема из устройства.

Smart Battery взаимодействует с другими устройствами (такими как SMBus Host и Smart Battery Charger) через два отдельных интерфейса взаимодействия:

- Первый использует линии CLOCK и DATA SMBus и является основным каналом взаимодействия между Smart Battery и другими SMBus устройствами. Smart Battery будет выдавать данные по запросу, отправлять информацию о процессе заряда зарядному устройству и оповещать о критических тревогах, когда (вычисленные или измеренные) параметры превышают допустимые пределы для конкретного типа Smart Battery.

- Второй требуемый интерфейс взаимодействия – это вторичный механизм сигнализирования «Safety Signal», который был ранее описан как «T-pin» на соединителе Smart Battery pack. Это выход переменного сопротивления из Smart Battery, который индицирует допустимость зарядки. Такой сигнал подразумевается как альтернативный метод, если SMBus становится неработоспособным. В основном, он используется зарядным устройством для обеспечения корректной зарядки. (За дополнительной информацией обратитесь к разделу 4.4.4 Safety Signal Hardware Requirements и Smart Battery Charger Specification).



Smart Battery Charger – это схема зарядного устройства, которая обеспечивает Smart Battery зарядным током и зарядным напряжением, удовлетворяющим запрошенным требованиям Smart Battery. Это позволяет батарее управлять своим циклом заряда. Опционально, Smart Battery Charger может не позволить Smart Battery поставлять энергию остальной части системы когда Smart Battery полностью заряжена и система подключена к AC сети, таким образом, продляя жизнь батарее.

SMBus хост представляет собой элемент электронного оборудования, который запитан от Smart Battery и который может взаимодействовать с ней. SMBus хост запрашивает информацию у батареи и затем использует ее в схеме управления энергоснабжения системы и/или использует её для обеспечения пользователя информацией о факте наличия или отсутствия батареи и характеристик батареи. SMBus хост также получает критические события от Smart Battery, когда та регистрирует проблему. В дополнение к тревожным сигналам, посылаемым в Smart Battery Charger, он получает тревожный сигнал об окончании разряда, оставшейся емкости ниже пользовательского порога и оставшееся время работы ниже пользовательского порога.

Хотя на рисунке выше SMBus показана как одна законченная шина, на самом деле она может быть сегментирована так, чтобы включать только те устройства, которым нужно взаимодействовать друг с другом. Например, SMBus хост, Smart Battery и Smart Battery Charger. Отдельные сегменты могут быть использованы для других SMBus устройств. Дополнительные Smart Battery могут быть добавлены с помощью Smart Battery Selector и/или Smart Battery System Manager для автоматической конфигурации SMBus сегментов.

**4.2 Определение программного обеспечения Smart Battery.**

Программный интерфейс разбит на три части: SMBus Host-to-battery, charger-to-battery и battery-to-charger or SMBus Host. Кроме того, здесь также обсуждается сигнализация об ошибках и обработка ошибок.

**4.2.1 SMBus Host To Smart Battery.**

Данный вид взаимодействия используется для получения данных, которые либо представляются пользователю, либо системе управления питанием. Пользователь может получать два типа данных от батареи: фактические данные и прогнозируемые данные. Фактические данные – это данные, которые могут быть измерены: температура, напряжение пакета, ток разряда или заряда, либо же это могут быть данные о характеристиках батареи, такие как тип химического элемента. Прогнозируемые данные – это данные, которые вычисляются на основе состояния батареи и ее характеристик, например, оставшееся время жизни батареи при текущей скорости расхода энергии. Прогнозируемые данные также могут быть рассчитаны на основе заданных параметров для предсказания времени работы батареи при некотором токе заряда или разряда, которые в настоящее время не измеряются батареей (см. AtRate функции). Кроме того, так как батарея имеет тактовый источник, информация может быть представлена как среднее значение на фиксированном интервале.

Система управления питанием может опрашивать драйвер устройства чтобы определить, нанесет ли это действие вред целостности системы. Например, раскрутка диска в приводе на завершающей фазе жизни батареи может привести к существенной просадке напряжения ниже допустимого предела, что приведет к системному сбою. Чтобы предотвратить это, драйвер устройства нуждается в информации от батареи, которая позволит ему принимать правильные решения. Если драйвер опрашивает батарею и находит, что доступного количества энергии недостаточно, он может попросить систему управления питанием выключить некритичных потребителей энергии, таких как фоновая подсветка LCD-экрана и попытаться повторить действие.

SMBus Host to Smart Battery взаимодействие выполняет следующие функции:

* Дает пользователю знание оставшегося времени жизни батареи.
* Сообщает пользователю как много времени займет заряд батареи.
* Позволяет батарее предоставить точную информации своим пользователям.
* Определяет в режиме реального времени требования питания.
* Позволяет управлять питанием на основе «реальной» информации, получаемой от батареи.
* Позволяет производителям батарей собирать информацию об использовании Smart Battery.
* Позволяет производителям электрически «клеймить» батареи на этапе производства.

**4.3 Обнаружение и сигнализация об ошибках.**

Спецификация SMBus определяет простую систему для обработки ошибок. Система ошибок разработана так, чтобы минимизировать количество трафика на шине SMBus и минимизировать сложность системы, необходимой для взаимодействия с Smart Battery. Как Smart Battery, так и SMBus Host ответственны за обнаружение и сигнализацию ошибок. Пожалуйста, обратитесь также к System Management Bus (SMBus) Specification, Version 1.1, 1998 за информацией о дополнительном механизме обнаружения ошибок (PEC).

**4.3.1 Обнаружение ошибок.**

Smart Battery ответственна за обнаружение ошибок в соответствии со спецификацией SMBus. Когда Smart Battery обнаруживает условие ошибки (такое как переполнение, опустошение, таймаут, не поддерживаемая или зарезервированная команда, недоступность данных, занятость или плохой размер) она сообщает хосту о том, что была обнаружена ошибка. Считается, что все функции обрабатываются Smart Battery без ошибок, кроме случаев, когда Smart Battery сигнализирует о том, что имела место быть ошибка. После обработки каждой функции, Smart Battery должна поместить соответствующий код ошибки в биты BatteryStatus(). (Примечание: включая «ОК» или «no errors detected»).

SMBus хост также ответственный за мониторинг ошибок SMBus, которые могут происходить, когда SmartBattery подключена к «живой» шине. Они могут происходить в момент установки батареи или извлечения батареи из системы. Хотя Smart Battery не может вмешиваться в транзакции SMBus за это время, действие вставки или удаления устройства из SMBus может привести к непреднамеренным ошибкам. После обнаружения вставки или удаления устройства хост SMBus должен сгенерировать условие START-STOP для сброса всех коммуникаций на сегменте шины, а затем повторить попытку любых передач, которые могли быть в процессе.

Вставка или удаление Smart Battery может быть обнаружено, когда «Safety Signal» переходит из или в высокоимпедансное состояние (>100k). Когда устройство SMBus, выступающее в роли ведущего, обнаруживает ошибку, оно должно попытаться вернуть шину в состояние ожидания, создав условие STOP.

**4.3.2 Сигнализация ошибок.**

Smart Battery сигнализирует хосту о том, что была обнаружена неисправимая ошибка используя то преимущество SMBus шины, которое заключается в требовании того, что бит подтверждения должен быть отправлен ведомым после передачи каждого байта. Когда Smart Battery не может выставить бит подтверждения, хост SMBus обязан генерировать условие STOP, тем самым вызывая преждевременное прекращение передачи. Это сигнализирует хосту SMBus о том, что произошла ошибка. Некоторые функции возвращают невалидные данные, когда вычисленное значение выходит за заданный диапазон, а затем сообщают OK или ошибку переполнения в младшем полубайте регистра BatteryStatus. Например, при чтении AverageTimeToFull когда батарея разряжена, функция может вернуть 0xFFFF но все еще сообщать код ошибки OK.

Smart Battery ВСЕГДА должна подтверждать свой собственный адрес. Невыполнение этого требования может привести к тому, что SMBus хост или Smart Battery Charger ошибочно предположат, что Smart Battery НЕ присутствует в системе, хотя «Safety Signal» может быть использован для детектирования присутствия Smart Battery в системе. Однако, обратите внимание что Smart Battery может отказаться от подтверждения какого-либо байта, следующего за адресом, если она занята или по каким-то иным причинам не может ответить. Если это происходит, запрос должен быть повторен.

Альтернативный метод сигнализации об ошибке полезен, когда Smart Battery возвращает данные мастеру в протоколе Read Word. В этом случае батарея не имеет возможности сигнализировать об ошибке, удерживая бит подтверждения (NB: так как протокол Read Word подразумевает, что этот бит выставляет мастер), так как мастер затребовал Read Word протокол для подтверждения факта передачи байтов от батареи. Если батарея обнаруживает ошибку, она может сигнализировать мастеру об этом, притянув линию данных или клока к низкому уровню на время, превышающее период таймаута, указанный в спецификации SMBus 1.10. После этого мастер обязан прерывать транзакцию и вернуть шину в неактивное состояние.

После каждой транзакции SMBus, направленной в адрес Smart Battery, последняя должна поместить соответствующий код ошибки в нижний полубайт регистра BatteryStatus(). Если транзакция была завершена успешно, коды ошибок должны быть очищены, что будет означать отсутствие ошибок. Таймаут и другие ошибки, не описанные одним из типов кодов ошибок, могут сообщаться как Unknown Error.

Smart Battery может сигнализировать об ошибке, модулируя Safety Signal. Эта техника использует Safety Signal в качестве вторичного канала для расширения связи SMBus в случае критических условий. Safety Signal может быть выставлен в значение, которое запрещает заряд батареи, если это не безопасно. Батарея может также имитировать удаление или повторную установку батареи, мгновенно выставляя сопротивление Safety Signal более 100кОм.

**4.3.3 обработка ошибок.**

Когда Smart Battery сигнализирует об ошибке, хост может использовать функцию BatteryStatus() для извлечения кода ошибки из Smart Battery, чтобы изучить природу ошибки. В случае, когда код ошибки равен OK, SMBus хост может допустить, что произошла нераспознанная ошибка на шине, которая не была обнаружена Smart Battery. Все функции, обрабатываемые Smart Battery, считаются безошибочными, если Smart Battery не сигнализирует хосту SMBus, что произошла ошибка, или если хост не детектировал ошибки. Когда либо хост, либо батарея сигнализируют об ошибке, хост может принять решение о повторной попытке совершить транзакцию немедленно, или позднее.

**4.3.4 Тайминги ошибок.**

Нет никакой подстраховки, что другой мастер SMBus может опросить Smart Battery немедленно после передачи данных между хостом и Smart Battery. Такое дополнительное чтение данных приведет к сбросу кодов ошибок (NB: от предыдущей транзакции) в BatteryStatus() на основе этой последней передачи.

Самый безопасный способ гарантировать, что чтение BatteryStatus() будет соответствовать самому последнему считанному значению – это выполнить чтение BatteryStatus() немедленно после чтения значения исходных данных. Это может быть выполнено путем выдачи условия SMBus START после условия SMBus STOP от предыдущей передачи. (Допустимое минимальное время между STOP и START условиями оговорено в спецификации SMBus). Поскольку мастеру требуется проверить, что шина простаивает время, как минимум, 50мкс, эта техника может предотвратить попытку получения управления шиной другим мастером.

**3.3 Характеристики Smart Battery.**

Smart Battery может присутствовать или не присутствовать в системе. Кроме того, она может быть установлена динамически, или удалена динамически, когда система запитана. Поэтому батарея должна проявлять предсказуемое поведение при установке в систему и/или при включении системы. Ниже приведено описание состояний батареи и описание действий, которые происходят в результате изменений состояния.

**4.4.1 Начальные условия.**

Когда Smart Battery используется в первый раз, должны быть предустановлены несколько значений:

|  |  |  |
| --- | --- | --- |
| **Функция (значение данных)** | **Начальное значение** | **Единицы изм.** |
| RemainingCapacityAlarm() | 10% от DesignCapacity() значения | мАч |
| RemainingTimeAlarm() | 10 | минуты |
| BatteryMode() | Bit 15: CAPACITY\_MODE=0  Bit 14: CHARGER\_MODE=0  Bit 13: ALARM\_MODE=0  Bit 9: PRIMARY\_BATTERY=0  Bit 8: CHARGE\_CONTROLLER\_ENABLED=0 | бит |
| BatteryStatus() | Bit 7: INITIALIZED=1 | бит |
| CycleCount() | Не волнует | десятичное число |

Примечание: CycleCount() может иметь небольшое значение (обычно менее 5) если производитель пакета батарей или сборщик произвел некоторые начальные испытания или кондиционирование упаковки.

**4.4.2 Smart Battery «On State».**

Smart Battery входит в состояние «On State» всякий раз, когда она обнаруживает, что линии CLOCK и DATA интерфейса SMBus переходят в высокий уровень. Батарея должна быть активна и способна взаимодейтствовать через SMBus в пределах 1мс с момента «подъема» линий. Батарея не может портить трафик на SMBus, однако акт физической вставки батареи может привести к непреднамеренному прерыванию связи.

Smart Battery не может начать вещание сообщений ChargingVoltage(), ChargingCurrent() или AlarmWarning() в SMBus Host или Smart Battery Charger по крайней мере первые 10 секунд после входа в состояние «On State».

Когда Smart Battery входит в состояние «On State», следующие значения должны быть переинициализированы.

|  |  |  |
| --- | --- | --- |
| **Функция (значение данных)** | **Начальное значение** | **Единицы изм.** |
| BatteryMode() | Bit 15: CAPACITY\_MODE=0  Bit 14: CHARGER\_MODE=0  Bit 13: ALARM MODE=0  Bit 9: PRIMARY\_BATTERY=0  Bit 8: CHARGE\_CONTROLLER\_ENABLED=0 | бит |

Smart Battery по умолчанию отключает внутренний контроллер заряда (если он существует), чтобы предотвратить возможную перегрузку источника питания в системах, в которых присутствует более одной батареи. Без этого умолчания есть вероятность, что для нескольких батарей одновременно может потребоваться больше мощности заряда, чем доступно в системе питания. Smart Battery по умолчанию действует как вторичная батарея для того, чтобы предотвратить большие объемы энергии, которые могут потенциально протекать между двумя основными батареями с разными уровнями заряда.

**4.4.3 Smart Battery «Off State».**

Smart Battery может войти в состояние «Off State» всякий раз, когда линии интерфейса SMBus CLOCK и DATA остаются в низком уровне более 2.5с. Smart Battery может перейти в это состояние за меньшее время, но ни в коему случае не менее чем за 250мс. Линии SMBus могут установиться в низкий уровень из-за того, что:

* Батарея удалена из системы.
* SMBus хост установил линии в низкий уровень чтобы сбросить батарею.
* Снято питание с SMBus шины (выключили систему).

**4.4.4. Требования к сигналу «Safety Signal» (Smart Battery Charger Interface).**

Smart Battery обязана предоставить дополнительный сигнал для безопасной зарядки. Этот «Safety Signal» также наиболее известен как «T-pin» или «Thermistor» на некоторых соединителях Smart Battery. «Safety Signal» - это выходной сигнал Smart Battery, который может быть использован Smart Battery Charger’ом (или другим устройством) для определения факта допустимости заряда. Этот сигнал представляет собой выход переменного сопротивления, которое измеряется между выводом «Safety Signal» и отрицательным выводом батареи. Схема, формирующая этот сигнал, может быть очень простой для некоторых типов химических элементов батарей (например, непосредственно термистор можно использовать для NiCd и NiMH элементов) или более гибкой для других типов химических элементов (например LiION).

Воможности Smart Battery Charger изменяются в зависимости от значения Safety Signal. Как того требует механизм безопасности, зарядник НЕ должен заряжать батарею, если он намерил сопротивление между выводом «Safety Signal» и землей в диапазоне от 425Ом до 3150Ом. NiMH батареи могут использовать термистор 103AT в качестве источника Safety Signal, который войдет в этот диапазон если он станет слишком горячим. Li-ion батареи могут использовать дискретные резисторы в качестве источника Safety Signal, который войдет в этот диапазон при аварии.

Валидные диапазоны Safeti Signal приведены ниже вместе с указанием о соответствующих возможностях зарядного устройства (пожалуйста, обратитесь к Smart Battery Charger Specification).

Примечания: 1) в таблице выше «Wake-up Charge» означает выдачу максимального количества заряда, которое может принять Smart Battery до возобновления связи на SMBus. Это количество заряда (максимальный ток и время) определяется спецификацией Smart Battery Charger в 100 мА (максимум) в течение от 140с до 210с. Удаление Smart Battery из зарядного устройства и вставка его обратно приведет к сбросу Wake-up Charge, который затем может быть повторен. 2) Понятие «Controlled Charge», использованное в таблице заключается в использовании зарядным устройством значений ChargingVoltage() и ChargingCurrent(), сообщаемых ему Smart battery.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Сопротивление Safety Signal, Rss**  **Ом** | **Charger**  **Status Bits** | **Описание** | **Wake-up**  **Charge** | **Controlled**  **Charge** | **Комментарий** |
| 0 < Rss < 575 | RES\_UR,  RES\_HOT | ниже диапазона | допускается в начальный момент времени | допускается | Зарядник может выполнять Wake-up Charge на время таймаута, но управляемый заряд разрешен. |
| 425 < Rss < 3150 | RES\_HOT | горячий | не допускается | не допускается | Сбой при заряде – зарядник не должен выдавать ток |
| 2850 < Rss < 31.5k | (none) | нормальный диапазон | допускается без ограничений | допускается | Зарядник может выполнять Wake-up Charge без ограничений, управляемый заряд разрешен. |
| 28.5k < Rss < 105k | RES\_COLD | холодный | допускается в начальный момент времени | допускается | Зарядник может выполнять Wake-up Charge только на время таймаута |
| Rss > 95k | RES\_OR,  RES\_COLD | выше диапазона | не допускается | не допускается | может быть использовано для детектирования (NB: съема) батареи. Зарядник не должен выдавать ток. |

**4.4.5 Требования к опросу и обновлению данных.**

Нет никаких ограничений по скорости (кроме тех, что установлены спецификацией SMBus) или частоты запросов данных Smart Battery. Непрерывный опрос на высокой скорости допускается, но не рекомендуется из-за ограничений пропускающей способности SMBus и доступности. Smart Battery может задержать любой запрос данных, удерживая линию CLOCK в низком уровне до 25мс. Это можно сделать для того, чтобы пересчитать запрошенное значение или извлечь данные из устройства хранения.

У Smart Battery также имеется возможность удерживать линию CLOCK в низком уровне до 35мс, чтобы вызвать таймаут и полностью отменить запрос. Кроме того, Smart Battery может NACK (не подтвердить) любой байт, следующий за его собственным адресом, чтобы прервать любую передачу.

Устройства, которые непрерывно ведут опрос Smart Battery, с высокой степенью риска могут пропустить AlarmWarning() и ChargingVoltage()/ChargingCurrent() сообщения от SmartBattery и поэтому должны считывать эти значения не реже раза в 10с для обеспечения надлежащего уведомления об этих условиях и значениях.

Данные Smart Battery должны обновляться с разумной частотой, чтобы обеспечивать систему управления питанием полезной информацией. Следует избегать длительных задержек между фактическими изменениями параметров батареи и обновлением значений. Как правило, значения данных должны обновляться при изменении парамтеров батареи, в зависимсоти от режима работы батареи (active или at rest) и эти обновления должны происходить в пределах 5с от соответствующих изменений параметров.

Обратитесь к функциям Smart Battery Current(), AverageCurrent(), Voltage() и Temperature() (разделы с 5.1.9 по 5.1.12) за конкретными требованиями ко времени измерения и обновления.

**5. Интерфейс Smart Battery.**