Using the bq33100

Host System Interaction

Contents

Overview

The bq33100 provides an array of data relating to the state of the Super Capacitors and the system via the SMBus Interface. This data can be used to provide overall system level status and enable a predictable maintenance schedule for in field service.

Review of the bq33100 Data Sheet [SLUS987] should be undertaken to ensure full understanding of the configuration options available in the device.

Hardware Interface

The data is accessed via the SMBus interface, SDA (pin14) and SCL (pin16) [See Appendix A – SMBus Electrical Characteristics for Electrical Characteristics of the SMBus Interface].

The bq33100 acts as a slave only with a fixed address of 0x16 and can mastered by an SMBus 1.1 or 2.0 or I^2C Master (See Appendix B - Main Differences Between SMBus and I2C) with the following data format:

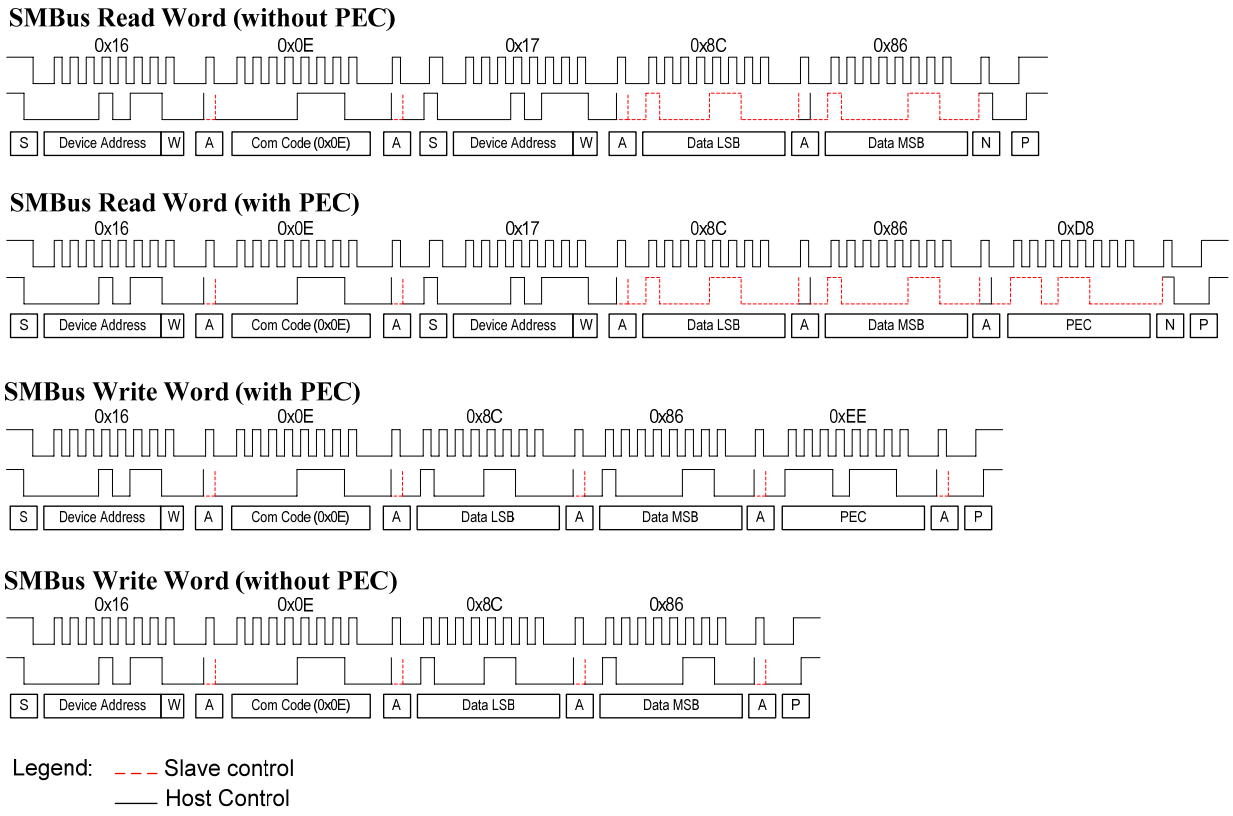


Figure 1: SMBus Protocol

Software Interface

The bq33100 data can be accessed through the SMBus Read Word function and can access the following array of data.

Standard Data Commands

SBS Cmd	Mode	Name	Format	Size in Bytes	Min Value	Max Value	Unit
0x00	R/W	ManufacturerAccess	hex	2	0x0000	0xffff	
0x08	R	Temperature	unsigned int	2	0	65535	0.1degK
0x09	R	Voltage	unsigned int	2	0	65535	mV
0x0a	R	Current	signed int	2	-32768	32767	mA
0x0b	R	ESR	unsigned int	2	0	65535	mΩ
0x0d	R	RelativeStateOfCharge	unsigned int	1	0	100	%
0x0e	R	Health	unsigned int	1	0	100	%
0x10	R	Capacitance	unsigned int	2	0	65535	F
0x14	R	ChargingCurrent	unsigned int	2	0	65534	mA
0x15	R	ChargingVoltage	unsigned int	2	0	65534	mV
0x3b	R	CapacitorVoltage5	unsigned int	2	0	65535	mV
0x3c	R	CapacitorVoltage4	unsigned int	2	0	65535	mV
0x3d	R	CapacitorVoltage3	unsigned int	2	0	65535	mV
0x3e	R	CapacitorVoltage2	unsigned int	2	0	65535	mV
0x3f	R	CapacitorVoltage1	unsigned int	2	0	65535	mV
0x40..0x4f	-	Reserved	-	-	-	-	-

Extended Data Commands

0x50	R	SafetyAlert	Hex	2	0x0000	0xffff	
0x51	R	SafetyStatus	Hex	2	0x0000	0xffff	
0x54	R	OperationStatus	Hex	2	0x0000	0xf7f7	
0x60	R/W	UnSealKey	Hex	4	0x000000 00	0xffffffff	
0x70	R/W	ManufurerInfo	String	31+1	-	-	

Some commands are broken down further into control functions or status reporting functions.

ManufacturerAccess (0x00)

This read- or write-word function provides Super Capacitor data to system along with access to bq33100 controls and security features. To use the ***ManufacturerAccess( )*** command simply send the sub command listed below as the data for the SBS Command 0x00.

Sub Cmd	Mode	Name	Description
0x0001	R	Device Type	Returns the IC part number.
0x0002	R	Firmware Version	Returns the firmware version.
0x0003	R	Hardware Version	Returns the hardware version

0x0004	R	DF Checksum	Generates a checksum of the full Data Flash (DF) array
0x0020	W	Seal	Enters Sealed mode with limited access to the extended SBS functions and data flash space
0x0021	R/W	Lifetime Enable	0 = Disables logging of lifetime data to non-volatile memory 1 = Enables logging of lifetime data to non-volatile memory
0x0022	R	IF Checksum	Returns the value of the Instruction Flash (IF) checksum
0x0023	W	Learn	This write function instructs the bq33100 to enter a capacitance learning cycle(Capacitance Update).
0x0024	W	Learn Value Reset	This write function instructs the bq33100 to reset the capacitance learned values (Capacitance) to initial default values.
0x0025	W	Learn Init	This write function instructs the bq33100 to enter a capacitance learning cycle(Capacitance Update) and update Initial values for Capacitance and ESR
0x0030	W	FAULT Activation	Drives the FAULT pin high
0x0031	W	FAULT Clear	Sets FAULT pin low
0x0032	W	Charge Level Nominal	Drives the CHGLVL0,1 pins low
0x0033	W	Charge Level A	Drives the CHGLVL0,1 pins high, low
0x0034	W	Charge Level B	Drives the CHGLVL0,1 pins low, high
0x0035	W	Charger Level Max	Drives the CHGLVL0,1 pins high
0x0037	W	Learn Load Activation	Drives the LLEN pin high (does not activate actual learning algorithm, see 0x0023)
0x0038	W	Learn Load Clear	Sets the LLEN pin low
0x0040	W	Calibration Mode	Places bq33100 into calibration mode
0x0041	W	Reset	bq33100 undergoes complete reset
Unseal Key	W	Unseal Device	Enables access to SBS and DF space
Extended SBS	R/W	*Extended SBS Commands*	Access to Extended SBS commands

SafetyAlert (0x50)

This read-word function returns indications of pending safety issues, such as running safety timers, or fail counters that are nonzero but have not reached the required time or value to trigger a *SafetyStatus* failure. These flags do not cause the FAULT pin to be set.

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
High Byte	CLBAD	RSVD	RSVD	RSVD	RSVD	OTC	CIM	OV
Low Byte	RSVD	RSVD	RSVD	RSVD	OCC	OCD	SCC	SCD

CLBAD 1 = Excessive capacitor leakage alert
OTC 1 = Charge overtemperature alert
CIM 1 = Capacitor voltage Imbalance permanent failure alert
OV 1 = Capacitor overvoltage alert
OCC 1= Overcurrent during charge alert
OCD 1= AFE overcurrent during discharge alert
SCC 1= AFE short circuit during charge alert
SCD 1= AFE short circuit during discharge alert

SafetyStatus (0x51)

This read-word function returns the status of the safety features. These flags do not cause the FAULT pin to be set unless the corresponding bit in ***FAULT Cfg*** is set.

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
High Byte	CLBAD	HFAIL	HWARN	HLOW	RSVD	OTC	CIM	OV
Low Byte	DFF	RSVD	AFE_C	WDF	OCC	OCD	SCC	SCD

CLBAD 1 = Excessive capacitor leakage fault
HFAIL 1 = Health fault
HWARN 1 = Health low warning
HLOW 1 = Health low indication
OTC 1 = Charge overtemperature fault
CIM 1 = Capacitor voltage Imbalance fault
OV 1 = Capacitor overvoltage fault
DFF 1 = Data Flash Fault permanent failure fault
AFE_C 1 = Permanent AFE Communications failure fault
WDF 1 = AFE Watchdog fault
OCC 1= Overcurrent during charge fault
OCD 1= AFE overcurrent during discharge fault
SCC 1= AFE short circuit during charge fault
SCD 1= AFE short circuit during discharge fault

OperationStatus (0x54)
This read-word function returns the current operation status

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
High Byte	RSVD	DSG	SS	FC	LTE	RSVD	RSVD	CB
Low Byte	LDTO	LCTO	LPASS	CL	RSVD	CFET	RSVD	RSVD

DSG Discharging
0 = bq33100 is in charging mode
1 = bq33100 is in discharging mode, relaxation mode, or valid charge termination has occurred
SS 1 = Sealed security mode
FC 1 = Fully Charged
LTE 1 = Lifetime data enabled
CB 1 = Capacitor voltage balancing in progress
LDTO 1 = Learning discharging phase time out
LCTO 1 = Learning charging phase time out
LPASS 1 = Learning complete and successful
CL 1 = Capacitance learning in progress
CFET 1 = Charge FET output ON

Host Data and FAULT Interaction

There are some key pieces of information that the host can use to help maintain system power integrity and enable predictive maintenance of the super capacitors.

Under normal operation with the Super Capacitor array fully charged the FAULT pin is low, ***SafetyAlert( )*** = 0, ***SafetyStatus( )***= 0 and ***OperationStatus( )*** = 0x7824 [DSG =1, SS=1, FC=1, LTE=1, CB=0, LDTO=0, LCTO=0, LPASS=1,CL=0, CFET=1]. As various operating conditions change the ***OperationStatus( )*** register will change to indicate the current status.

It is recommended that the **Fault** data flash location be configured to enable the FAULT pin activation when there is a fault with the Super Capacitor system indicated by a bit being set (1) in the ***SafetyStatus( )*** register. When the corresponding bit in **Fault** and ***SafetyStatus( )*** are both set then the FAULT pin is triggered.

The bq33100 can indicate that the Super Capacitor array is reaching the end of its usable life within the application. Specifically the ***HLOW***, ***HWARN*** and ***HFAIL*** bits in ***SafetyStatus( )***. The remaining bits within ***SafetyStatus( )*** indicate that there is a temporary error in the Super Capacitor system

The FAULT pin is intended to be an interrupt to the host indicating a change in fault status. The host should then read the ***SafetyStatus( )*** register and the ***OperationStatus( )*** registers to determine the current state of the Super Capacitor system and take appropriate action.

Alternatively the host can poll the bq33100 the ***SafetyStatus( )*** and ***OperationStatus( )*** registers and take appropriate action when needed.

Appendix A – SMBus Electrical Characteristics

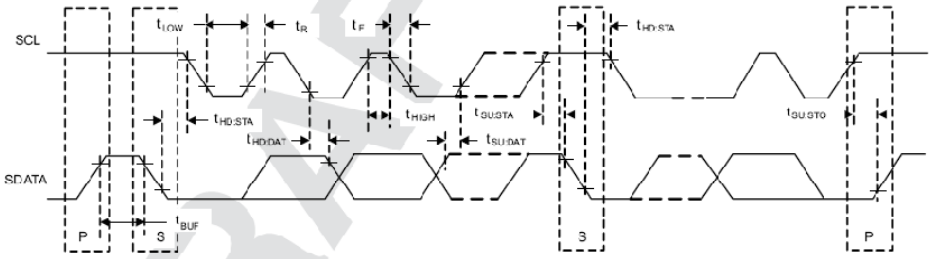
SMBus

Typical values stated where T_A = 25°C and VCCPACK = VCC = 14.4V, Min/Max values stated where T_A = -40°C to 85°C and VCCPACK = VCC = 3.8V to 25V (unless otherwise noted)

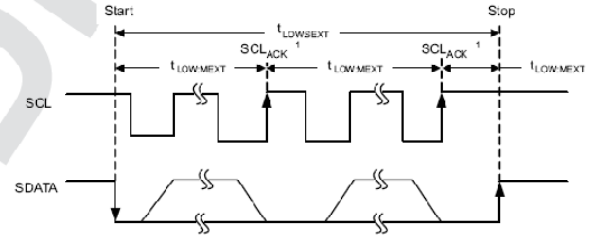
PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
f_{SMB}	SMBus operating frequency	Slave mode, SCL 50% duty cycle	10		100	kHz
f_{MAS}	SMBus master clock frequency	Master mode, no clock low slave extend		51.2		kHz
t_{BUF}	Bus free time between start and stop		4.7			µs
$t_{HD:STA}$	Hold time after (repeated) start		4.0			µs
$t_{SU:STA}$	Repeated start setup time		4.7			µs
$t_{SU:STO}$	Stop setup time		4.0			µs
$t_{HD:DAT}$	Data hold time	Receive mode	0			ns
		Transmit mode	300			
$t_{SU:DAT}$	Data setup time		250			ns
$t_{TIMEOUT}$	Error signal/detect	See (1)	25		35	ms
t_{LOW}	Clock low period		4.7			µs
t_{HIGH}	Clock high period	See (2)	4.0		50	µs
$t_{LOW:SEXT}$	Cumulative clock low slave extend time	See (3)			25	ms
$t_{LOW:MEXT}$	Cumulative clock low master extend time	See (4)			10	ms
t_F	Clock/data fall time	See (5)			300	ns
t_R	Clock/data rise time	See (6)			1000	ns

(1) The bq33100 times out when any clock low exceeds $t_{TIMEOUT}$
(2) t_{HIGH}, Max, is the minimum bus idle time. SCL = SDA = 1 for t > 50 µs causes reset of any transaction involving bq33100 that is in progress. This specification is valid when the NC_SMB control bit remains in the default cleared state (CLK[0]=0). If NC_SMB is set then the timeout is disabled.
(3) $t_{LOW:SEXT}$ is the cumulative time a slave device is allowed to extend the clock cycles in one message from initial start to the stop.
(4) $t_{LOW:MEXT}$ is the cumulative time a master device is allowed to extend the clock cycles in one message from initial start to the stop.
(5) Rise time $t_R = V_{ILMAX} - 0.15$) to ($V_{IHMIN} + 0.15$)
(6) Fall time $t_F = 0.9V_{DD}$ to ($V_{ILMAX} - 0.15$)

SMBus Timing



SMBus $t_{TIMEOUT}$



(1) SCL_{ACK} is the acknowledge-related clock pulse generated by the master.

Appendix B[1] - Main Differences Between SMBus and I2C

The major differences between I2C and SMBus fall into several categories including electrical, timing, protocols and operating modes.

DC Specifications for SMBus and I2C

Both I2C and SMBus are capable of operating with mixed devices that either have fixed input levels (such as Smart Batteries) or their input levels are related to VDD. When mixing devices, the I2C specification defines the VDD to be 5.0 Volt +/- 10% and the fixed input levels to be 1.5 and 3.0 Volts.

Version 1.1 of SMBus instead of relating the bus input levels to VDD, it defines them to be fixed at 0.8 and 2.1 Volts. This SMBus specification allows for bus implementations with VDD ranging from 3 to 5 Volts +/- 10%. SMBus has relaxed in this version the initial requirement for fixed input levels of 0.6 and 1.4 Volts, in order to reduce the cost of SMBus compliant devices. Devices compliant with the 1.0 specification of SMBus will still operate with a version 1.1 SMBus. In addition they will be ready to operate with 2 Volt SMBus implementations in the future.

A second difference in the DC parameters between I2C and SMBus is in the power consumption of the bus.

SMBus was designed to accommodate extremely low power consumption devices, such as the control circuitry within a Smart Battery. These devices have limited current sinking capabilities and a low power consumption bus is essential for maintaining communications without draining the battery of a mobile computer. As a result, SMBus sets more stringent DC requirements than I2C. One of the main differences is the IOL specification for VOL = 0.4 Volts. SMBus devices are required to sink a minimum of 100 uA as opposed to 3mA specified for I2C devices for the same VOL.

A third difference is in the specification of the maximum leakage current for each device connected to the bus. I2C specifies the maximum leakage current to be 10 uA. SMBus version 1.0 specified maximum leakage current of 1 uA. Version 1.1 of the SMBus specification relaxes the leakage requirements to 5 uA, in order to reduce the cost of testing of SMBus devices.

Finally, SMBus does not specify a maximum bus capacitance. Instead it specifies the $I_{PULLDOWN}$ maximum of 350 uA. Bus capacitance can be calculated taking into consideration the maximum rise time and $I_{PULLDOWN}$.

The following table lists the main differences among the DC parameters for I2C and SMBus.

Symbol	Parameter	Std I2C mode device		Fast I2C mode device		SMBus device		Units
		MIN	MAX	MIN	MAX	MIN	MAX	

[1] [Reference: System Management Bus Specification, Revision 1.1, December 11, 1998]

Symbol	Parameter	Std I2C mode device		Fast I2C mode device		SMBus device		Units
		MIN	MAX	MIN	MAX	MIN	MAX	
VIL	Fixed input level	-0.5	1.5	-0.5	1.5	-0.5	0.8	V
	VDD related input level	-0.5	0.3VDD	-0.5	0.3 VDD	N/A	N/A	V
VIH	Fixed input level	3.0	VDDmax+ 0.5	3.0	VDDmax+0 .5	2.1	5.5	V
	VDD related input level	0.7VDD	VDDmax+ 0.5	0.7VDD	VDDmax+0 .5	N/A	N/A	V
VHYS	VIH-VIL	N/A	N/A	0.05VDD	-	N/A	N/A	V
VOL	VOL @ 3mA	0	0.4	0	0.4	N/A	N/A	V
	VOL @ 6mA	N/A	N/A	0	0.6	N/A	N/A	
	VOL @ 350uA	N/A	N/A	N/A	N/A	-	0.4	
IPULLUP		N/A	N/A	N/A	N/A	100	350	uA
ILEAK		-10	10	-10	10	-5	5	uA

Table 1: DC Parameter Comparison Between Standard I2C and SMBus Devices

Timing specifications differences of I2C and SMBus

There are differences in the timing specifications between I2C and SMBus. As in the case of DC specification, proper understanding of the parameters is needed in order to combine reliably I2C with SMBus devices.

1. SMBus defines a minimum bus clock frequency F_{SMB} of 10 KHz. I2C does not specify any minimum bus frequency. Besides maintaining effective bus throughput, this SMBus specification parameter can be used as a simple way to detect a bus idle condition (in addition or in lieu of detecting each STOP condition) as well as to implement bit timeout.
2. Maximum clock frequency for SMBus is defined at 100 KHz. I2C provides two modes of operation. The STANDARD MODE up to 100 KHz and the FAST-MODE up to 400 KHz.
3. SMBus defines a clock low time-out, $T_{TIMEOUT}$ of 35 ms. I2C does not specify any timeout limit.
4. SMBus specifies $T_{LOW:SEXT}$ as the cumulutative clock low extend time for a slave device. I2C does not have a similar specification.
5. SMBus specifies $T_{LOW:MEXT}$ as the cumulative clock low extend time for a master device. Again I2C does not have a similar specification.
6. SMBus defines both rise and fall time of bus signals. I2C does not.

The SMBus time-out specifications do not preclude I2C devices co-operating reliably on the SMBus. It is the responsibility of the designer to ensure that I2C devices are not going to violate these bus timing parameters.

Other differences

ACK and NACK usage

There are the following differences in the use of the NACK bus signaling:

- In I2C, a slave receiver is allowed not to acknowledge the slave address, if for example is unable to receive because it's performing some real time task. SMBus requires devices to acknowledge their own address always, as a mechanism to detect a removable device's presence on the bus (battery, docking station, etc.)
- I2C specifies that a slave device, although it may acknowledge its own address, some time later in the transfer it may decide that it cannot receive any more data bytes. The I2C specifies, that the device may indicate this by generating the not acknowledge on the first byte to follow.

Besides to indicate a slave device busy condition, SMBus is using the NACK mechanism also to indicate the reception of an invalid command or data. Since such a condition may occur on the last byte of the transfer, it is required that SMBus devices have the ability to generate the not acknowledge after the transfer of each byte and before the completion of the transaction. This is important because SMBus does not provide any other resend signaling. This difference in the use of the NACK signaling has implications on the specific implementation of the SMBus port, especially in devices that handle critical system data such as the SMBus host and the SBS components.

SMBus protocols

Each message transaction on SMBus follows the format of one of the defined SMBus protocols. The SMBus protocols are a subset of the data transfer formats defined in the I2C specifications. I2C devices that can be accessed through one of the SMBus protocols are compatible with the SMBus specifications.

I2C devices that do not adhere to these protocols cannot be accessed by standard methods as defined in the SMBus and ACPI specifications.