

SMBus[™] Communication with Ultralife SMARTCIRCUIT[®] Batteries

A White Paper Analysis

Author: Michael Manna, *Director of Engineering, Government and Defense*

Date: 10/28/2008



SMART! CIRCUIT[®]



[page intentionally left blank]

ABSTRACT

This white paper has been written to address common questions that result from use of the Ultralife Batteries SMARTCIRCUIT[®] family of battery products—the most common questions being those that regard implementation of the communication protocol, *SMBus*.

OVERVIEW

SMBus V1.1 is a derivative of the popular I²C[®] bus developed by Phillips, which is a simple-to-use, three-wire (ground, clock, and data) serial-bus architecture. Though the I²C bus has many popular uses, it is primarily used for low-to-moderate serial speed communication on board-level designs that require a peripheral-to-processor communication link such as remote-temperature sensors.

I²C versus SMBus

Several differences exist between the standard I²C bus and SMBus, the largest being the maximum speed. While SMBus maximum speed is 100 kHz, I²C can be as fast as 400 kHz or higher. As with most network architectures, I²C will negotiate down to slower clock speeds to communicate with lower-speed devices such as SMBus.

Another key difference between the two busses is addressing scheme. That is, SMBus does not truly allow addressable items as with most I²C devices. Instead, SMBus assigns id 11 (0x0B) to batteries on the bus, and other devices get the next-available ID on the bus as communication is established. This being the case, addresses for battery IDs can occur at random when multiple batteries are on the same bus. This leads to difficulty in physically identifying which battery is communicating from which address.

Making addressing difficulties worse, the addresses will change dynamically every time batteries are removed from the device or the device is fully discharged and powered down. This is not an entirely bad situation, as there are several multiplexer (MUX) devices available from various vendors which can help identify each data and clock line. Using a MUX will enable the designer to identify specific batteries repeatably and reliably without worry of battery addressing changes.

OPERATION

Once the SMBus communication protocol is established, information between the host device and the battery can be exchanged. “Smart” batteries have several broadcast modes, such as charge/current request and error/alert notifications. The fields typically available to a host device are defined in the Smart Battery Data (SBData) Specification, V1.0. This specification lists the parameters required by smart batteries to be compliant.

Several important parameters used by the host device to monitor battery condition and status are voltage, current, remaining capacity, and absolute state-of-charge. Further, the battery can communicate remaining run time to the host to warn of impending power loss due to battery discharge through low-battery condition alarms. Typically, the reported remaining capacity and run-time information has temperature and discharge rate compensation to account for battery performance and internal impedance.

DESIGN CONSIDERATIONS

A noteworthy design consideration is that the use of SMBus and I²C buses requires a low-capacitance path. If these communication protocols are utilized across a cable, distance must be kept to a minimum, which typically less than six feet. Using shielded cable or twisted-pair cable may assist in providing more reliable communication. Several devices are available that can extend and “square” off the communication

waveforms to fight high capacitance or longer cable runs and thus create reliable communication links.

Another overlooked design consideration is communicating with a battery that is used in a series application such as shown in Figure 1. This is an issue because data and clock for SMBus are relative to the ground point of the circuit, and batteries utilized in series applications shift the ground of the higher-voltage section. That is, the actual ground point is equal to the voltage of the lower-battery section (10-16.8V DC on a UBBL10/13) and continually shifts throughout the battery discharge. This requires the use of an isolated input for proper communication or a shorting path will exist to system ground.

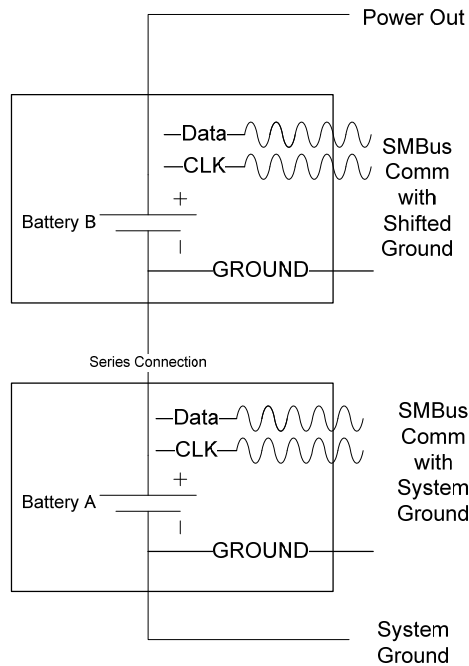


Figure 1: Batteries Shown in Series Configuration

Several companies make optical and inductive isolators for I²C and SMBus, including NXP Semiconductors and Analog Devices, Inc. It is good practice to add some ESD protection and filtering to the battery communication interface.

Most batteries will tolerate 3.3V to 5V pull-up communication levels. A specific design may require this voltage to vary, especially if mixing I²C and SMBus on a single communication bus. This scenario may require a voltage-level translator in the design if operating on a lower voltage bus (i.e., 1.8V).

ADDITIONAL INFORMATION

Helpful links for further SMBus and related information are listed below:

www.smartbattery.org/

sbs-forum.org/

www.analogdevices.com

www.linear.com

www.nxp.com

www.ti.com

For additional information on Ultralife products and services, please visit:

ULTRALIFE[®]

Corporate Websites

www.ultralifecorporation.com

www.ultralifebatteries.com

www.mcdowellresearch.com

www.stationarypower.com

www.rpspowersystems.com

www.redblackcomm.com

We. Are. Power.®

This paper is presented as a helpful guide for users and integrators of the Ultralife SMARTCIRCUIT® batteries. Information contained in this document is intended for reference only, and can change at any time. Ultralife assumes no responsibility for the content or material provided and/or how it may be interpreted or used. Ultralife suggests only experienced designers undertake battery communication projects. Please contact Ultralife for further assistance as required.

SMBus™ is a trademark of Intel Corporation.

I²C® is a registered trademark of Philips Corporation.

ULTRALIFE® and SMARTCIRCUIT® are registered trademarks of Ultralife Corporation.

All other trademarks included herein are the property of their respective owners.