

# I<sup>2</sup>C Battery Gauge Interface Using Altera MAX Series

2014.09.22

AN-493



Subscribe



Send Feedback

Altera<sup>®</sup> MAX<sup>®</sup> II, MAX V, and MAX 10 devices can be used in this example application.

This design example demonstrates how to use a supported Altera device as an I<sup>2</sup>C battery gauge interface. The end product is a low-cost intelligent device that improves communication between the battery gauge and the host system.

## Related Information

- [Design Example for MAX II](#)  
Provides the MAX II design files for this application note (AN 493).
- [Design Example for MAX 10](#)  
Provides the MAX 10 design files for this application note (AN 493).
- [Power Management in Portable Systems Using MAX II CPLDs](#)
- [MAX II CPLD Design Guidelines](#)

## Battery Gauge on the I<sup>2</sup>C Bus

A battery gauge is a device which continuously monitors the state of a battery's capacity and displays it to the user in a simple form. Often in embedded systems, the status of the battery supply is monitored remotely. Communication between the embedded system and battery gauge can be facilitated through an industry standard two-wire I<sup>2</sup>C bus and an interfacing device. Because of its low overhead, the two-wire communication link minimizes the number of traces required on the PCB to monitor the power source.

A supported Altera device can easily serve as the interfacing device. The I<sup>2</sup>C interface allows the supported Altera device to access readings from the battery gauge remotely. An I<sup>2</sup>C serial interface consists of a data line (SDA) and a clock line (SCL). Both the lines are bidirectional and are pulled up to the V<sub>CC</sub> of the system. This bus is also a common bus for any other I<sup>2</sup>C-compliant device in the system.

## Using a Supported Altera Device to Interface with a I<sup>2</sup>C Battery Gauge

In this implementation, the battery gauge acts as slave to the supported Altera device and has a software programmable address. At any time, the supported Altera device acting as the master can read the exact state of battery charge through the battery gauge interface. The design uses an inexpensive battery gauge device, such as the Maxim DS2745.

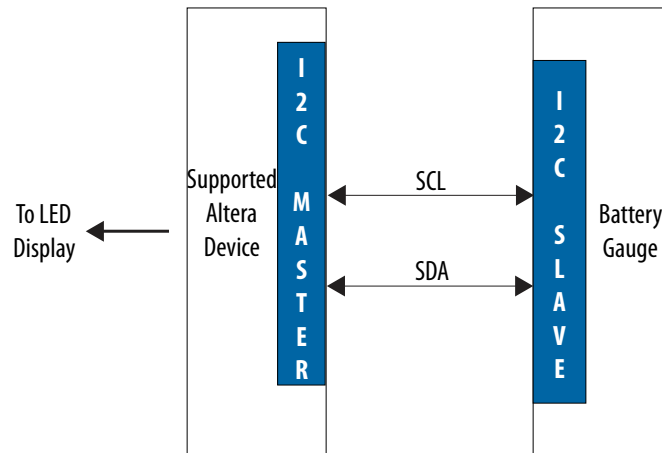
Because the design is made for illustration, it does not implement a complete I<sup>2</sup>C master. This design, therefore, does not accommodate multimaster capability and is only meant to demonstrate the capability of the supported Altera device to read the DS2745 battery gauge via an I<sup>2</sup>C interface.

© 2014 Altera Corporation. All rights reserved. ALTERA, ARRIA, CYCLONE, ENPIRION, MAX, MEGACORE, NIOS, QUARTUS and STRATIX words and logos are trademarks of Altera Corporation and registered in the U.S. Patent and Trademark Office and in other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders as described at [www.altera.com/common/legal.html](http://www.altera.com/common/legal.html). Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

ISO  
9001:2008  
Registered

**Figure 1: I<sup>2</sup>C Interface between Battery Gauge and the Supported Altera Device**

The bidirectional lines SCL and SDA form the input and output of the system.

**Table 1: I<sup>2</sup>C Interface Pin Description**

Lists the I<sup>2</sup>C signals used in the design.

Signal	Purpose	Direction
SCL	I <sup>2</sup> C Clock	Output
SDA	I <sup>2</sup> C Serial Data	Bidirectional

In order to retrieve the status of the battery from the gauge, the master initiates the communication cycle by sending a Start condition on the bus. The Start condition, which consists of pulling the SDA line low while SCL is high, is followed by a 7-bit slave address and a write command (90h). After receiving these signals, the slave sends back an acknowledgment to the master.

**Figure 2: Master and Slave Communications**

S	Device Address (90h)	ACK	Memory Address (0h)	ACK	Sr	Device Address (91h)	Gauge Output (MSB)	ACK	Gauge Output (LSB)	ACK	P
1-bit	8-bit Address with write	1-bit	8-bit	1-bit	1-bit	8-bit Address with write	8-bit	1-bit	8-bit	1-bit	1-bit

S: Start      ACK: Acknowledgement

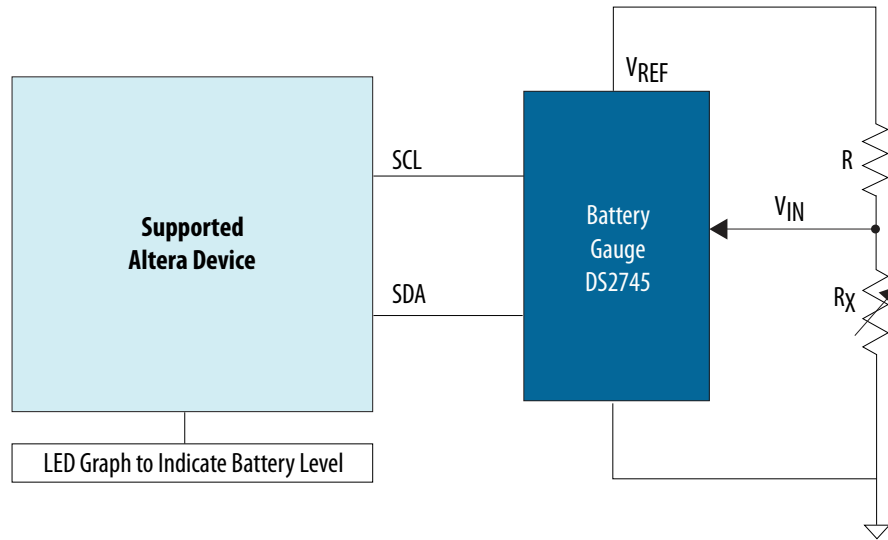
P: Stop      Sr: Repeat Start

Master Write       Slave Write

After receiving the acknowledgment (battery gauge DS2745), the master sends the memory address (0Ch), which corresponds to the voltage register on the slave. The DS2745 acknowledges this memory address, and then the master generates a Repeat Start (Sr). After the Repeat Start, the supported Altera device (master) sends the device address again, followed by a read, this time (91h). The DS2745 acknowledges it as the same. The battery data is transferred on the SDA line to the master in the next two 8-bit read

operations; the MSB is sent first followed by the LSBs. The master sends an acknowledgment after every 8 bits that are received. A Stop (P) condition is generated by the master by leaving the SDA line high when SCL is high after the read is complete. The slave returns Data in two's complement form, which is converted to a bar form for display.

**Figure 3: I<sup>2</sup>C Battery Gauge Demonstration Scheme**



**Related Information**

**Low-Cost I2C Battery Monitor**

Provides more information about the Maxim DS2745.

## Implementation

The detailed description of the implementation is based on the MAX II devices. This application can also be implemented in MAX V and MAX 10 devices.

This design example can be implemented with an EPM240 device or any other MAX II device and a I<sup>2</sup>C battery gauge device such as the Maxim DS2745. This design communicates with the Maxim DS2745 battery gauge through the I<sup>2</sup>C bus and displays the constantly updated voltage readings without requiring any external stimulus.

The voltage status returned by the DS2745 battery gauge is 11-bit wide and is in 2's complement form. The voltage readings are converted into a viewer-friendly form with eight voltage magnitude levels by the MAX II device. These are subsequently displayed with the help of an LED array with eight LEDs.

The following details the implementation of this design example on the MDN-B2 demo board.

**Table 2: Design Example Using the MDN-B2 Demo Board**

EPM240G Pin Assignments			
Signal	Pin	Signal	Pin
SCLK	pin 39	SDA	pin 40

EPM240G Pin Assignments			
Signal	Pin	Signal	Pin
led_level[0]	pin 76	led_level[1]	pin 75
led_level[2]	pin 74	led_level[3]	pin 73
led_level[4]	pin 72	led_level[5]	pin 71
led_level[6]	pin 70	led_level[7]	pin 69

Assign unused pins as **input tri-stated** in the Quartus® II software. You must also enable the **Auto Open Drain** setting on the SCLK and SDA pins. To do this, on the Assignments menu, click **Settings** and then select **Analysis and Synthesis Settings** to enable the **Auto Open-Drain** setting. These settings are followed by a compilation cycle.

This design example has been implemented in Verilog and successful operation has been demonstrated using the MDN-B2 demo board, as described in this document.

## Design Notes

To demonstrate this design on the MDN-B2 Demo Board, follow these steps:

1. Turn on the power to the demo board (using slide switch SW1).
2. Download the design on to the MAX II device through the JTAG header JP5 on the demo board and a conventional programming cable (ByteBlaster™ II or USB-Blaster™). Keep SW4 on the demo board pressed while starting the programming process. After programming, turn off the power and remove the JTAG connector.
3. Mount the DS2745 battery gauge module (supplied with the MDN-B2 demo board) on the header JP3 of the demo board. Make sure that the red mark on the connector meets pin #1 on JP3.
4. Turn on the power to the MDN-B2 demo board.
5. Observe that the red LED array on the demo board vary with changing voltages on the battery gauge module. This voltage can be varied by changing the preset on the battery gauge module.

## Acknowledgments

Design example adapted for Altera MAX 10 FPGAs by:

Orchid Technologies Engineering and Consulting, Inc.

Maynard, Massachusetts 01754

TEL: 978-461-2000

WEB: [www.orchid-tech.com](http://www.orchid-tech.com)

EMAIL: [info@orchid-tech.com](mailto:info@orchid-tech.com)

## Document Revision History

Date	Version	Changes
September 2014	2014.09.22	Added MAX V and MAX 10 devices.
December 2007	1.0	Initial release.