MAX1730x/MAX1731x Software Implementation Guide

UG6807; Rev 0; 2/19

Abstract

This document covers the design details of the host software for the MAX17301-MAX17303/MAX17311–MAX17313 fuel gauges.

Table of Contents

List of Figures

List of Tables

Introduction

This application note describes the basic operations the host software needs to interface with the MAX17301–MAX17303/MAX17311–MAX17313. There are many additional features not covered here that are listed in the data sheet. For any additional questions, contact Maxim Integrated.

Table 1. Device Types

DEVICE TYPE	CELL COUNT	COMMUNICATION INTERFACE
MAX1730x	Single Cell	I²C
MAX1731x	Single Cell	1-Wire®

To distinguish the device type, read the DevName register (0x21). The low nibble indicates 0x1 for MAX173x1, 0x2 for MAX173x2, and 0x4 for MAX173x3. The 1-Wire/I²C distinction is made by a device present on one interface and not the other.

Initialization

At IC power-up, the fuel gauge retrieves the battery model parameters from the nonvolatile memory without any interaction from the host software.

Optionally, the host can change some of the auxiliary functions, such as alerts on voltage, current, state of charge (SOC), or temperature. If the nonvolatile programmed values are acceptable, then the host only needs to take action on the alert outputs. If the host desires different thresholds, the host can write to the VAlrt_Th, TAlrt_Th, SAlrt_Th, or IAlrt_Th registers as necessary. Refer to the MAX1730x/MAX1731x data sheet for register details.

Useful Registers to Read

Table 2 lists the registers that the host finds useful in most applications.

Table 2. Useful Registers in Most Applications

ADDRESS	REGISTER NAME	PURPOSE/CONTENTS
0x000	Status	Alert status and chip status.
0x005	RepCap	Reported remaining capacity.
0x006	RepSOC	Reported state of charge.
0x01A	VCell	Cell voltage for a single cell.
0x01B	Temp	Temperature.
0x01C	Current	Battery current.
0x011	TTE	Time to empty.
0x020	TTF	Time to full.
0x0D9	ProtStatus	Fault status of the protection functionality.
0x1A8	nBattStatus	Permanent battery status information.

Other applications might have more requirements and might want to read additional registers such as the ones listed in Table 3.

Table 3. Additional Registers

ADDRESS	REGISTER NAME	PURPOSE/CONTENTS
0x0D8	Cell1	Direct cell measurement.
0x134, 0x135	Temp1, IntTemp	Individual temperature measurements from the thermistor and internal die temperature.
0x0B9	AgeForecast	Projected total cycles the battery lasts until full capacity is below the target "dead" battery capacity.
0x007	Age	Percentage of full capacity compared to the design capacity.
0x017	Cycles	Number of cycles the battery has been used.
0x0DD (AtRate=0x004)	AtTTE	After writing AtRate, hypothetical time to empty based on the AtRate value.
0x010	FullCapRep	Full capacity based on present discharge conditions.
0x0BE	TimerH	Time since first power-up.
0x008, 0x009, 0x00A	MaxMinVolt, MaxMinTemp, MaxMinCurrent	Maximum and minimum values for each measurement since the last life log.
0x014	RCell	Calculated battery resistance.
0x019, 0x01D, 0x016	AvgVCell, AvgCurr, AvgTA	Average of voltage, current, or temperature.
0x0B2	VRipple	Filtered difference of average and instantaneous values of VCell.

Alerts Management

The default configuration of this fuel gauge is to automatically clear the alert after the alert condition is no longer present. The host can set the alerts to be "sticky," and make them persist until the host clears the alert. This setting can be changed in nonvolatile memory to permanently alter the behavior to "sticky" mode by changing the nConfig register. This change can be made by the pack maker or the end application if the nonvolatile memory is not locked. Alternatively, it can be changed by the host software in the Config register to only change it until the fuel gauge resets or when the host wants to disable this feature. The Config register details to make this change are provided in Table 4.

Table 4. Config Register (01Dh) Format

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	SS	TS	VS	0	PBen	0	0	SHDN	COMMSH	0	ETHRM	FTHRM	Aen	Bei	Ber

The SS, TS, and VS bits set the corresponding alerts to be "sticky." The alerts are cleared by clearing the appropriate alert bit in the Status register.

An example to set the low SOC alert to 5% is as follows:

1. Set SAlrtTh register to 0xFF05. 0xFF__ indicates that the high alert is disabled. 0x__05 sets the low alert to 5%.
2. Set the Config.Aen bit to 1.
3. When the SOC decreases below 0x0500, the alert pin asserts, and the status register indicates that the SOC alert has occurred.
4. To disable further alerts on the low SOC until the battery is charged, set the SAlrtTh register to 0xFF00 to disable low SOC alerts.

Authentication

This IC has an authentication feature to prevent clone batteries. The host writes a challenge to the IC and reads out a response. The host validates this response by either doing a parallel calculation with the same secret or using predefined challenge and response pairs.

Method 1: The Host Knows the Secret and Calculates a Valid Output to Check Against the Battery Pack Response

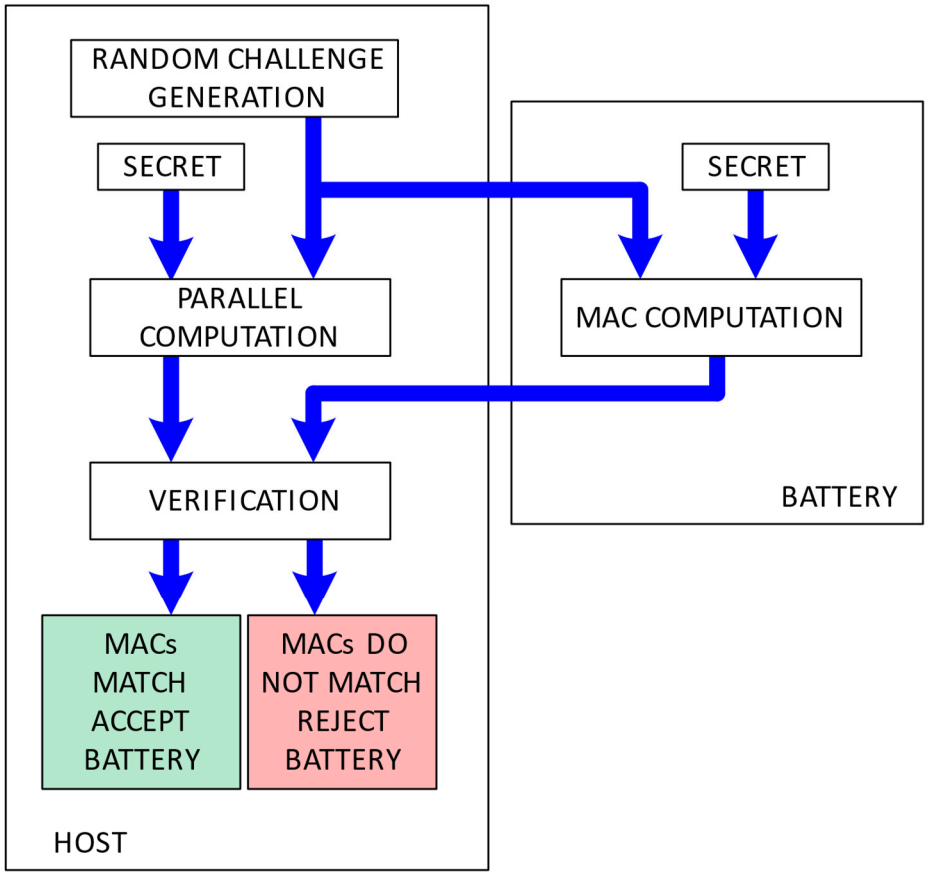

Figure 1. Parallel calculation with the same secret.

Method 2: The Host Does Not Know the Secret and the Challenge and Response Pairs are Predefined for the Host

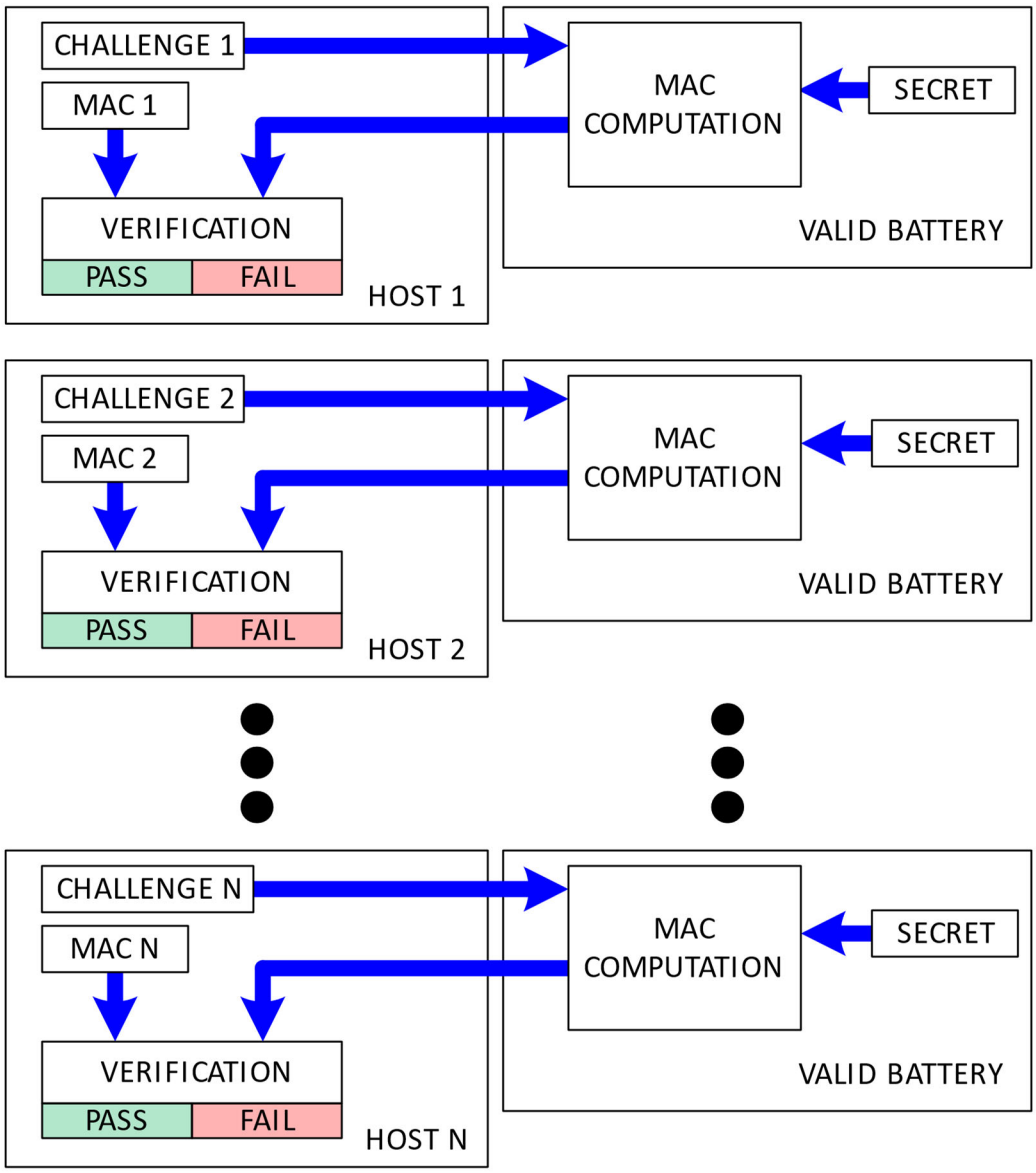

Figure 2. Predefined challenge and response pairs.

To authenticate the battery, the host writes the challenge to memory addresses 0x0C0 to 0x0C9, and then sends the Compute MAC without ROM ID command by writing to register 0x060 with the value 0x3600. The fuel gauge calculates a response after t_{SHA}, and the host can read it from spaces 0x0C0 to 0x0CF. The returned MAC computation is compared against the calculated or stored response, and the authenticity is verified.

If the host knows the secret, the Compute MAC with ROM ID command can be used for more security. Write to register 0x060 with the value 0x3500 to send the Compute MAC with ROM ID command. The returned MAC computation is available after t_{SHA}.

Revision History

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	2/19	Initial release	—