Keywords: battery fuel gauging, fuel gauging ICs, battery characterization, battery management, lithium ion batteries

APPLICATION NOTE 6642

ACCURATE FUEL GAUGING WITHOUT BATTERY CHARACTERIZATION USING MODELGAUGE M5 EZ ALGORITHM

Abstract: For conventional fuel gauges, battery characterization is mandatory to obtain acceptable performance due to the large distribution of the electrical properties presented by a wide array of lithium-ion chemistries available in the market. Even for batteries with the same chemistry such as the popular lithium cobalt oxide (LiCoO2), different form factors change the impedance in the batteries that traditionally makes battery characterization a necessary step. ModelGauge™ m5 EZ fuel gauge configuration is an innovative, adaptive fuel-gauging approach, delivering excellent fuel-gauge performance across many different battery chemistries, capacities, and charge voltages. The accuracy of the underlying robust adaptive algorithm has been proven on Maxim's exhaustive battery characterization database. ModelGauge m5 EZ eliminates the need for battery characterization, saving resources and time.

Introduction

With the Internet of Things (IoT) and connected devices being the major focus of new developments, a vast majority of new devices need to be operated from batteries. Traditionally, new products differentiate in the hardware itself. The recent trend, though, is that the key features that distinguish from competition are reflected in software. Compared to the way that organizations have developed products—in big teams with dedicated specialized engineers for every development task—the emphasis is now increasingly on keeping software teams small and agile in order to bring new products to market as quickly as possible.

Traditional methods of fuel gauging require a power or battery specialist on the team to work with the fuel-gauge vendor to find a suitable model that can be used with their battery. This often involves characterizing the battery under various load and temperature conditions, which can easily take a couple of weeks to fully characterize and model the batteries. This "old-fashioned" approach no longer fits the above-mentioned way of small, agile, and software-focused development teams.

Thus, a modern fuel-gauging solution that fits today's requirements, facilitates (in addition to the obvious need for accuracy over aging) ease-of-use and short time to market.

ModelGauge™ m5 EZ is the latest evolution of fuel-gauging algorithms that combines those features. It enables a quick and accurate fuel-gauging solution, without dedicated battery characterization.

This application note elaborates on the algorithm and how it implements different features. The section "Measurement and Simulation Results" provides more insight in the accuracy that is achieved using this approach. We also introduce a device family that implements the algorithm on a system-on-chip (SoC).

Working Principal of ModelGauge m5 EZ Algorithm

The patented ModelGauge m5 algorithm (Wortham, et al., 2014) (Wortham, 2013) uses real-time electrical measurements and converts these into usable SOC% (state of charge) and other battery information. The algorithm has multiple mechanisms of desensitizing the errors due to model mismatch with the actual cells in use. These mechanisms also desensitize any errors in the electrical measurements from having adverse effects on the SOC% output. In addition, there are several adaptive mechanisms that help the fuel gauge learn about the battery characteristics and improve its accuracy.

The ModelGauge m5 algorithm combines the short-term accuracy and linearity of a coulomb counter with the long-term stability of a voltage-based fuel gauge. The core of the algorithm combines the open-circuit voltage (OCV) state estimation with the coulomb counter. The OCV value of Li+ cell correlates to the SOC%, and this relationship persists largely independent of the age of the cell (see **Figure 1**).

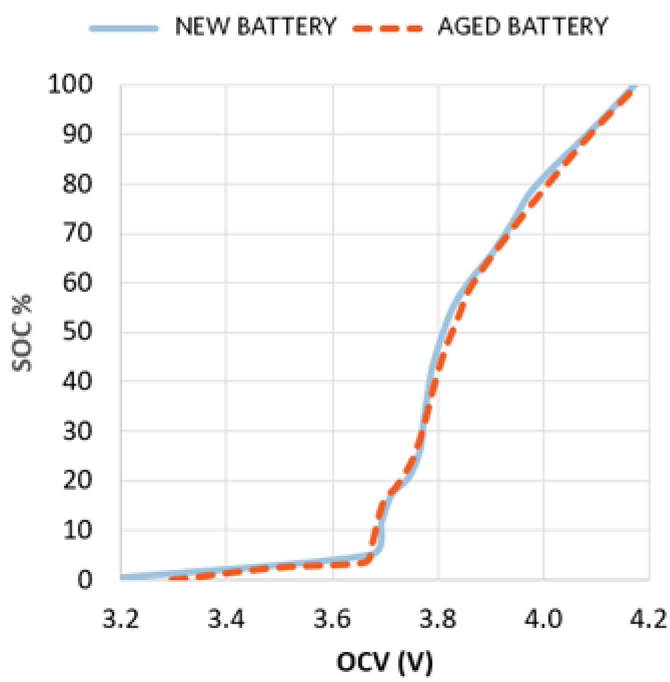

Figure 1. SOC% vs. OCV of a battery does not change with age.

As the cell cycles during the application, this process of traversing up and down through this curve largely desensitizes any local errors resulting from any model to cell mismatch. At the start, when the cell is first connected to the fuel gauge IC, the OCV state estimation is weighted heavily compared to the coulomb count output. As the cell is cycled in the application, coulomb counter accuracy improves and the mixing algorithm alters the weighting so that the coulomb counter result is dominant. From this point forward, the algorithm switches to servo mixing.

Servo mixing provides a fixed magnitude continuous error correction to the coulomb count, up or down, based on the direction of error from the OCV estimation. This allows differences between the coulomb count and OCV estimation to be corrected quickly. The resulting output from the mixing algorithm does not suffer accumulation drift from current measurement offset error and is more stable than a stand-alone OCV estimation algorithm (see **Figure 2**).

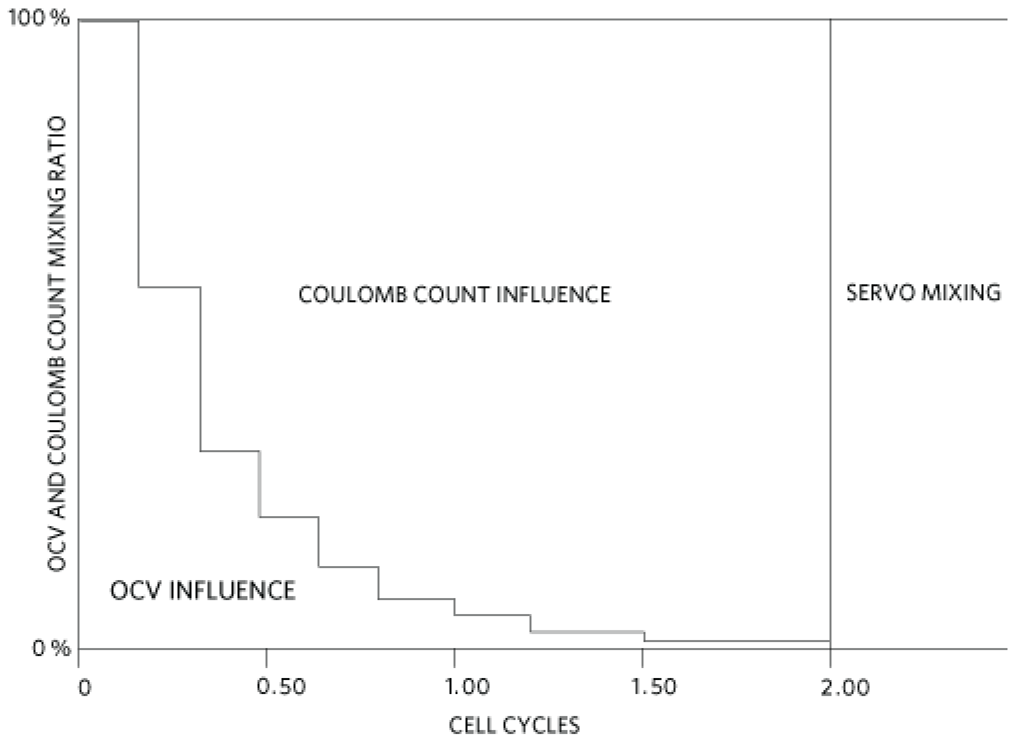

Figure 2. Voltage and coulomb count mixing.

This correction to the coulomb counter takes place continuously while the application is active and when it is in standby condition. In practical terms, this means that the coulomb counter corrections happen more than 200,000 times per day—in tiny steps that are almost invisible to the user. These corrections happen when the battery is under load, as well as at no-load condition, regardless of whether the cell is relaxed or not, and this is a significant advantage over other competing algorithms.

As the temperature and discharge rate of an application changes, the amount of charge available to the application also changes. The ModelGauge m5 algorithm distinguishes between remaining capacity of the cell and remaining capacity of the application, and reports both results to the user.

The algorithm periodically makes internal adjustments to cell model and application information to remove initial error and maintain accuracy as the cell ages. These adjustments always occur as small corrections to prevent instability of the system and prevent any noticeable jumps in the fuel gauge outputs. Learning occurs automatically without any input from an external host controller. In addition to estimating the battery's SOC, the ICs observe the battery's relaxation response and adjust the dynamics of the voltage fuel gauge.

The ModelGauge m5 algorithm includes a feature that guarantees the fuel gauge output converges to 0% as the cell voltage approaches the empty voltage. As the cell voltage approaches the expected empty voltage, the IC smoothly adjusts the rate of change of SOC % so that the fuel gauge reports 0% at the exact time that the cell voltage reaches empty. This prevents unexpected shutdown or an early 0% SOC reported by the fuel gauge. This also provides an additional mechanism of desensitizing the SOC % error from any model mismatch errors.

Measurement and Simulation Results

Maxim has developed a vast battery database consisting of cell characteristics and behavior over a variety of test conditions similar to the customers' use cases. This allows Maxim to validate any new improvements in the fuel-gauge algorithm, by running it on the real data collected previously. Using this data, Maxim analyzed the performance over hundreds of batteries of various sizes and plotted a histogram of the results in **Figure 3**.

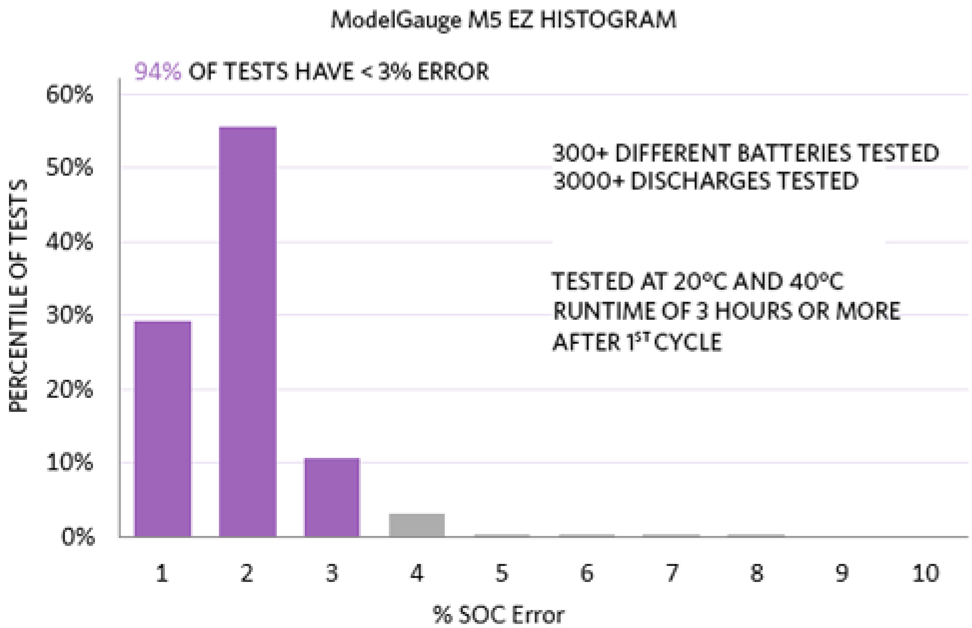

Figure 3. ModelGauge m5 EZ SOC error statistics on different batteries and discharges.

This shows that more than 94% of test cases at room temperature and above have less than 3% SOC error. These test cases do exclude certain battery types that are known to be quite different in terms of OCV versus SOC% table, compared to the more conventional and popular chemistries.

Figure 4 is a histogram that shows a comparison of the EZ model versus a "tuned" custom model, plotted as percentile of test cases versus the error bucket into which they fall. While the tuned model indeed places a higher number of cases in the 1% bucket, the aggregate of all test cases up to 3% error shows that the EZ model covers 95% while the custom mode covers 97% of the test cases. Considering the extra effort, resources, and time required to prepare a custom model, the EZ configuration approach is an attractive alternative.

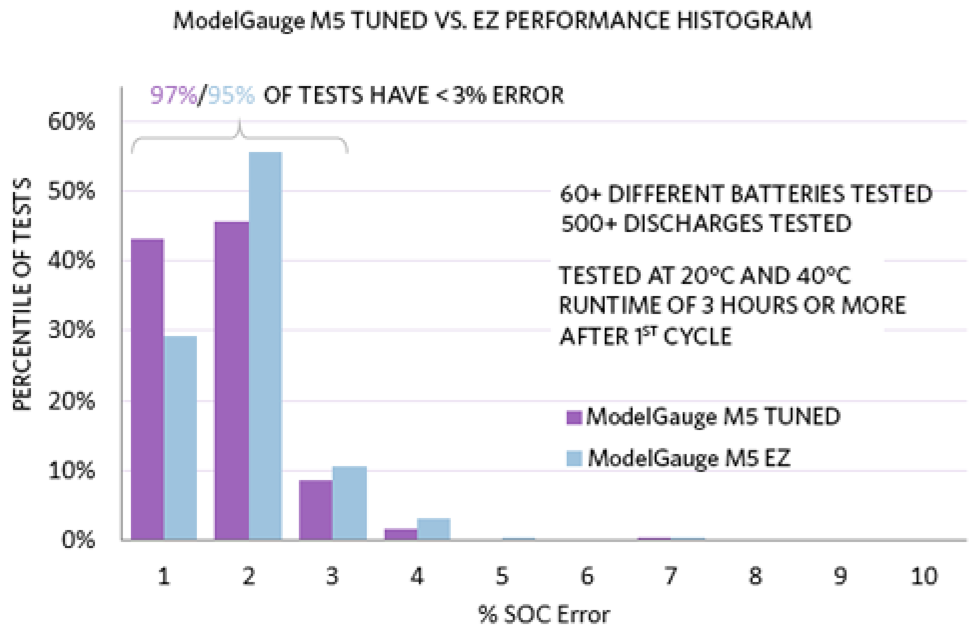

Figure 4. ModelGauge m5 EZ error compared to custom-tuned model.

Another way to look at this is to compare the EZ versus custom-tuned model at specific error budgets allowed in the system design. **Figure 5** shows how they compare with < 3% error budget and < 5% error budget.

Instead of simply looking at the worst-case error everywhere between 0% and 100% SOC, look at the error near empty (e.g., 10%), where accurate fuel gauging really matters. If the battery is around 50% state, and the fuel gauge is indicating 40% or 60% (10% error), nothing bad is likely to happen, as no critical power-management decisions are taken at that point. However, when the battery is at 10% and the fuel gauge indicates 5% SOC, then most likely the system is going to shut down prematurely, and the battery will not be utilized fully.

On the other hand, if the battery is at 5% and the fuel gauge indicates 10% SOC, then it is likely that the system will crash unexpectedly without the benefit of a graceful planned shutdown. Both result in poor user experience—the former results in a shorter runtime than expected, while the latter results in an abrupt shutdown that is very annoying for the user.

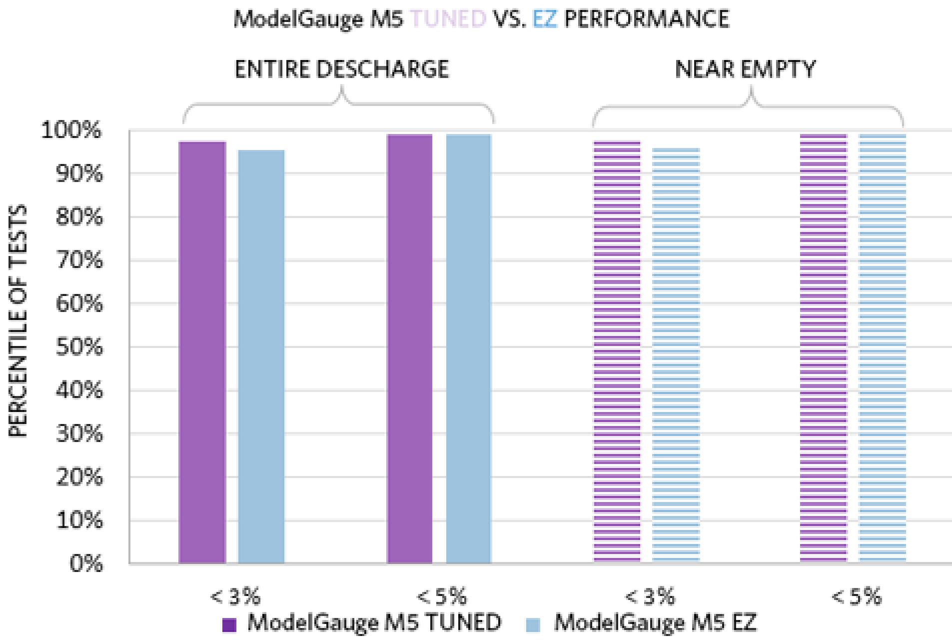

Figure 5. ModelGauge m5 EZ compared to custom-tuned model based on system error budget.

If the application has more demanding requirements and also needs good accuracy at cold temperatures (0° Celsius), then a similar analysis shows that the results are nearly the same for SOC error budget of < 5%. Thus, for a large category of applications, the simplicity of implementing the EZ configuration performance becomes a game-changer for new product development.

Silicon Implementation

Maxim has implemented the algorithm described above in the MAX17201, MAX17205, MAX17211, and MAX17215 device family. **Figure 6** shows the family block diagram.

The ICs automatically compensate for cell aging, temperature, and discharge rate, and provide accurate SOC in milliampere-hours (mAh) or percentage (%) over a wide range of operating conditions. The ICs provide accurate estimation of time-to-empty and time-to-full, Cycle+™ age forecast, and three methods for reporting the age of the battery: reduction in capacity, increase in battery resistance, and cycle odometer.

The ICs provide precision measurements of current, voltage, and temperature. The battery pack's temperature is measured using an internal temperature measurement and up to two external thermistors supported by ratiometric measurements on auxiliary inputs. The devices can provide alerts by detecting a high or low voltage, current, temperature, or SOC. The ICs also contain two programmable, fast overcurrent comparators that allow spikes in system current to be detected and warn the system to make appropriate adjustments to prevent such conditions that could cause the battery to crash abruptly. Both comparators have programmable threshold levels and programmable debounced delays.

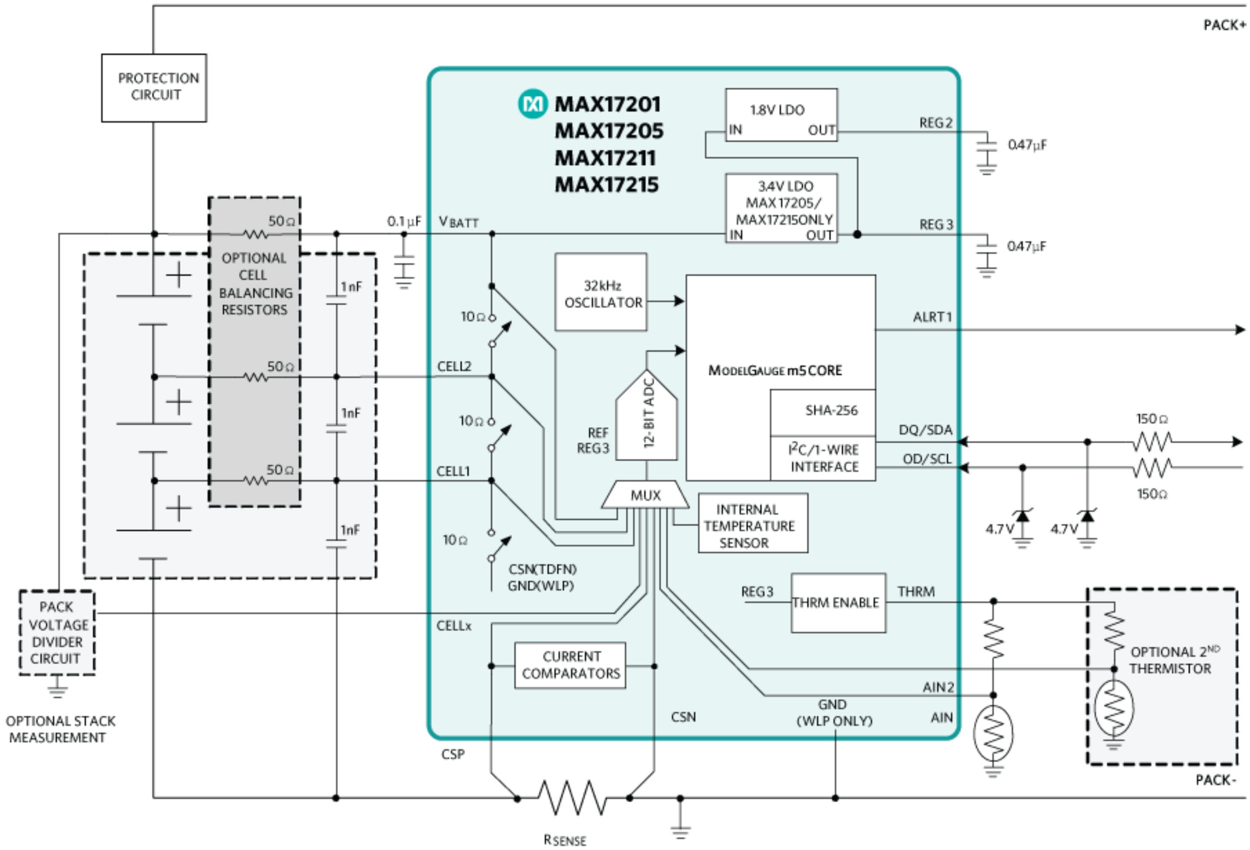

Figure 6. Block diagram of ICs implementing ModelGauge m5 EZ.

Conclusion

With its accuracy and ease-of-use, ModelGauge m5 EZ is a well-suited fuel-gauging algorithm with respect to accuracy. It also fits today's software and time-to-market focused design constraints. This application note examined the functionality of the algorithm, shared measurement results confirming its performance, and introduced a silicon implementation of the algorithm.

A similar version of this application note was first published in the proceedings issue of the conference '24. Entwicklerforum Batterien und Ladekonzepte' ('24th Developers Forum Batteries and Charging Concepts') of Design & Elektronik, Germany, in February 2017.

References

- Maxim Application Note 5037: Characterizing a Battery for Use with a Fuel Gauge
- Maxim MAX17201 Stand-Alone ModelGauge m5 Fuel Gauge with SHA-256 Authenticatio
- **Wortham Jason Allen and Ghaseminejad Parviz** ENHANCED VOLTAGE-BASED FUEL GAUGES AND METHODS [Patent] : US 8,643,331 B1. - USA, 02 04, 2014.
- **Wortham Jason Allen** MODEL-BASED BATTERY FUEL GAUGES AND METHODS [Patent] : US 8,502,504 B1. - USA, 08 06, 2013.

Related Parts		
MAX17201	Stand-Alone ModelGauge m5 Fuel Gauge with SHA-256 Authentication	Samples
MAX17205	Stand-Alone ModelGauge m5 Fuel Gauge with SHA-256 Authentication	Samples
MAX17211	Stand-Alone ModelGauge m5 Fuel Gauge with SHA-256 Authentication	Samples
MAX17215	Stand-Alone ModelGauge m5 Fuel Gauge with SHA-256 Authentication	Samples

More Information
For Technical Support: https://www.maximintegrated.com/en/support
For Samples: https://www.maximintegrated.com/en/samples
Other Questions and Comments: https://www.maximintegrated.com/en/contact

Application Note 6642: https://www.maximintegrated.com/en/an6642
APPLICATION NOTE 6642, AN6642, AN 6642, APP6642, Appnote6642, Appnote 6642