



Step-by-step Design Example: 1s

Battery Management Deep Dive Training

October 17-19, 2017

Onyx Ahiakwo

TI Information – Selective Disclosure. Battery Management Deep Dive 2017

1



Agenda

- **Introduction**
- **4 predesign steps**
 - Determine system requirements
 - Consider usage conditions
 - Choose a suitable gauge
 - Define test conditions
- **Bqstudio and EVM**
- **Calibration**
- **Chem id identification**
- **Data flash configuration**
- **Learning cycle and golden file extraction**
- **Evaluation Tests**
- **Conclusion**

TI Information – Selective Disclosure. Battery Management Deep Dive 2017

Introduction

- Choice and correct configuration of an ideal gauge to meet an application's needs is critical to the overall success of the end equipment
- This presentation discusses a design example of a single cell battery gauge system
- The various key considerations are outlined and should act as a guide for other typical systems.

4 predesign steps

- Determine system requirements
- Consider usage conditions
- Choose a suitable gauge
- Define test conditions

Step 1: Determine system requirements

- The design example in this design walk-through will be a cell-phone application

Key Questions

- What is your accuracy target? Ex. 1% error or 25% error
- Pack-side (smart battery) or system-side?
- Power consumption of the gauge?
- Temperature range?
- Load profile
 - Constant or dynamic?
 - Long periods in standby? (Are you worried about self-discharge errors?)
 - Always charging or discharging? (Never gets a chance to relax?)
 - Battery always stays topped off? (Never gets a learning cycle?)
- How long is the battery expected to be used?
 - Do you want accuracy even after the battery is well aged?
- Gauge reporting
 - Do you only want to read State of Charge (SOC in %)?
 - Do you want capacity info (mAh)?
 - Do you want Time To Empty (TTE) or Time To Full (TTF) predictions?

Step 2: Consider usage conditions

- What factors impact usable capacity?
 - Temperature
 - Low temperature → higher impedance → less usable capacity
 - Current range
 - High current → more IR drop in cell → less usable capacity
- What conditions could make a gauge inaccurate?
 - Partial cycles
 - Never allowed to learn faded capacity
 - Coulomb counter gets off track
 - Dynamic loads
 - Long storage or standby time
- How will my battery age?
 - Why batteries degrade:
 - High cycle counts
 - Long calendar life
 - High charge rates kill cells
 - High charge voltage kills cells
 - High temperatures (even in storage) kill cells
 - What happens to old batteries:
 - Impedance goes up exponentially (result: less usable capacity)
 - Capacity fades (even for low discharges)

Step 2: Choose a suitable gauge

- Typical features
 - Reporting: SOC, Remaining Capacity, Time-To-Empty/Full,
 - Compensation: Can it compensate predictions for different loads (light/heavy) and different temperatures?
 - Self-discharge estimation: Can it compensate for internal self-discharge of the cell?
 - Aging: Can it compensate for or accurately gauge an aged battery?
 - Pack-side or host-side: Host side gauges might be introduced to an unexpected old or new battery. Can it handle it?

Step 2: Choose a suitable gauge cont'd

- Suitability of various algorithms
 - Voltage based algorithm $SOC = f(VBAT)$
 - May be ok for very light loads and narrow temperature ranges.
 - No worries about cell self-discharge since that will be reflected in voltage.
 - Can determine only SOC (relative).
 - Example issue: % of what?
 - » 50% of a new battery may give you 1 hour of run time.
 - » 50% of an old battery may give you only 15 minutes!
 - Even worse performance if voltage profile of your cell is very flat (ex. LiFePO4)
 - ERROR could be 50-100%, especially as cells age.
 - Coulomb counting algorithm $Q = \int i dt$
 - Not influenced by distortions in voltage measurements.
 - Current integration hardware is very accurate.
 - Modeling for new batteries could achieve < 3-10% error.
 - Poor performance as cells age.
 - Cannot account for cell self-discharge except for modeling.
 - Learning cycle required to periodically update total capacity.
 - Impedance Track algorithm
 - Best accuracy under all conditions, even for aged cells
 - Can account for self-discharge and low-level drain by the system
 - Pack-side and host-side versions available

Discrete single cell gauge

Performance

Performance

bq27320

System/Pack
CEDV+
Power: 118/23/1uA
15p CSP

bq27542/546

Pack side
IT Algorithm
Power: 118/23/1uA
12p QFN/ 15p CSP

bq27520/510

System side
IT Algorithm
Power: 118/23/1uA
15p CSP/12p QFN

Flash Based – fully configurable

ROM Based – plug & play with few parameters to configure

bq27220

System/Pack
Pre-programmed selectable CEDV
Ext Rsense
Ext Temp sense
Power: 50/9/0.6uA
9p CSP/12p QFN

bq27411

Pack side
Fixed profile
Ext R_{SENSE}
Power: 93/21/0.6uA
12p QFN

bq27421/441

System side
Fixed profile
Int R_{SENSE}
Power: 93/21/0.6uA
9p CSP/12p QFN

bq27426

System side
Pre-programmed selectable IT profile
Ext R_{SENSE}
Ext Temp Sense
Power: 50/9/0.6uA
9p CSP



Production



Development



Sampling

Step 4: Define test conditions

- What conditions do you want to test?
 - Nominal usage
 - Specific system operation scenarios
 - Corner cases (ex. High rate at low temperature)
- Define conditions for testing
 - Temperature?
 - Load rate & profile?
 - Test new cells then age and re-test?
 - Good way to age quickly is to cycle at high rates and high temp

Bqstudio and EVM

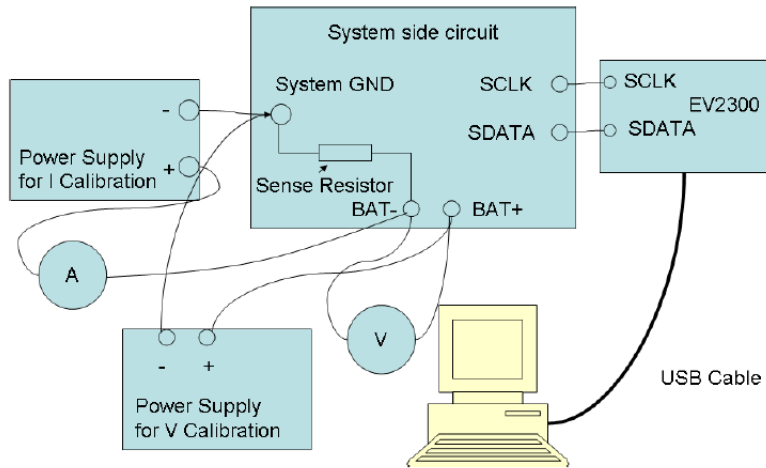
-bq27542/6 is the gauge choice

-bqstudio and its EVM is used for gauge evaluation and configuration

The screenshot displays the BqStudio software interface, version 1.3.73. The main window shows the 'Data Memory' section, which contains a table of data memory contents. The table lists various parameters and their values, units, and addresses. The parameters are organized into sections: Calibration, Configuration, Security, System Data, OCV Tables, Rb Tables, Ra Tables, Registers, HDQ, and Power. The 'Data Memory' section is currently selected, showing a list of parameters such as Charging Voltage, Charge Termination, and Temperature. The table columns include Name, Private, Value, Unit, Subclass ID, Data Length, Block Number, Block Offset, and Native Units. The 'Data Memory' section is currently selected, showing a list of parameters such as Charging Voltage, Charge Termination, and Temperature. The table columns include Name, Private, Value, Unit, Subclass ID, Data Length, Block Number, Block Offset, and Native Units.

Name	Private	Value	Unit	Subclass ID	Data Length	Block Number	Block Offset	Native Units
Charging Voltage		4400	mV	0x22	2	0	0	mV
Charge Termination								
Taper Current		200	mA	0x24	2	0	0	mA
Min Taper Capacity		150	mAh	0x24	2	0	2	mAh
Taper Voltage		100	mV	0x24	2	0	4	mV
Current Taper Window		40	s	0x24	1	0	6	s
TCA Set %		-1	%	0x24	1	0	7	%
TCA Clear %		98	%	0x24	1	0	8	%
FC Set %		-1	%	0x24	1	0	9	%
FC Clear %		98	%	0x24	1	0	10	%
DDatEOC Delta T		5.0	°C	0x24	2	0	11	0.1°C
JEITA								
T1 Temp		0	°C	0x27	1	0	0	°C
T2 Temp		10	°C	0x27	1	0	1	°C
T3 Temp		45	°C	0x27	1	0	2	°C
T4 Temp		50	°C	0x27	1	0	3	°C
T5 Temp		60	°C	0x27	1	0	4	°C
Temp Hys		1	°C	0x27	1	0	5	°C
T2-T3 Chg Voltage		4400	mV	0x27	2	0	6	mV
T2-T3 Chg Voltage		4400	mV	0x27	2	0	8	mV
T3-T4 Chg Voltage		4400	mV	0x27	2	0	10	mV
T4-T5 Chg Voltage		4400	mV	0x27	2	0	12	mV
T2-T3 Chg Current		50	%	0x27	1	0	14	%
T2-T3 Chg Current		80	%	0x27	1	0	15	%
T3-T4 Chg Current		80	%	0x27	1	0	16	%
T4-T5 Chg Current		80	%	0x27	1	0	17	%
Registers								
Pack Configuration		2031	Flag	0x40	2	0	0	Flag
Pack Configuration B		82	Flag	0x40	1	0	2	Flag
Pack Configuration C		38	Flag	0x40	1	0	3	Flag
Pack Configuration D		17	Flag	0x40	1	0	4	Flag
Filter	Private	20	Num	0x40	2	0	6	Num
DQ Scale	Private	1024	Num	0x40	2	0	8	Num
HDQ								
Host Interrupt Tries		3	Num	0x40	1	0	5	Num
Power								
Flash Update OK Voltage		2300	mV	0x44	2	0	0	mV
Sleep Current		15	mA	0x44	2	0	2	mA
Cal Inhibit Temp Low	Private	5.0	°C	0x44	2	0	4	0.1°C
Cal Inhibit Temp High	Private	45.0	°C	0x44	2	0	6	0.1°C
Sleep Minimum Time	Private	30	s	0x44	1	0	8	s

Gauge calibration



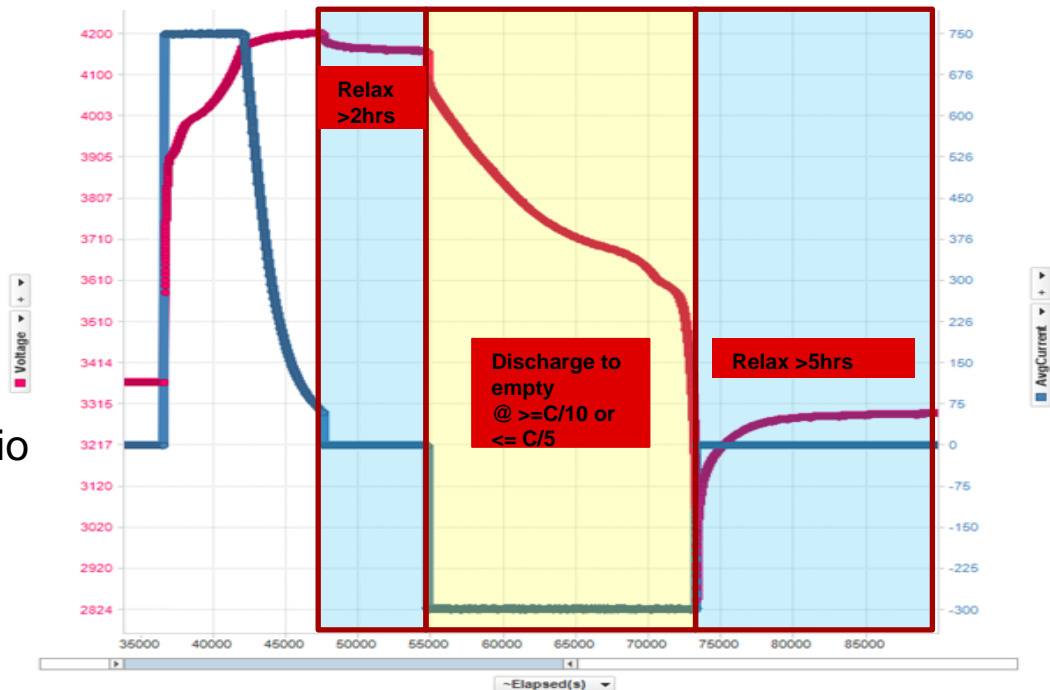
Calibrate the voltage, current , temperature measurements, cc offset and board offset

- Set up the environment as shown
- Use bqstudio to perform V,I,T and cc-offset and board offset calibration

Chem id identification using gpcchem

Identify the chem id of the battery

- Perform a relax-discharge-relax test while logging V,I,T with bqstudio
- Upload results to online tool gpcchem
- Program the best chemid returned on device using bqstudio



Data flash configuration

Configure the data flash parameter prior to learning. At minimum configure the following

- Taper current. Values between $C/10$ and $C/20$ should be used.
- Discharge current threshold. This should be less than taper current
- Charge current threshold. This should be less than taper current.
- Quit current. This determines the gauge's relaxation state. This value should be less than dsg and chg current thresholds and less than or equal to $C/20$.
- Design capacity: Rated Nominal capacity of the cell
- Design voltage: Rated Nominal voltage of the cell
- Charge voltage for the different temperature levels
- Terminate voltage: minimum voltage stated in cell data sheet that the cell can be discharged to

Learning cycle and golden file extraction

Learning cycle

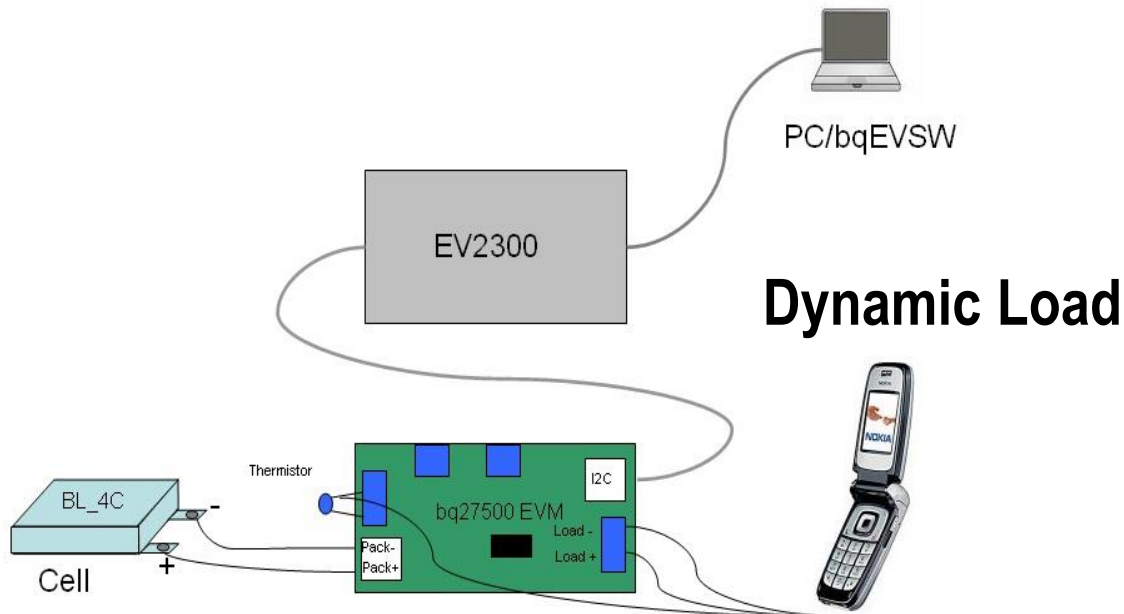
- Issue a IT enable command (0x21). Then Issue a reset command (0x41)
- Discharge to empty (terminate voltage) using a constant current value between C/5 and C/7
- Rest for 5 Hours
- Charge the battery to full(charge voltage specified in DS of cell) and make sure you taper to a value below the taper current in DF
- Rest for 2 hours (Qmax will update at this point). Update status will go from 04 to 05
- Discharge to empty using the same discharge rate as before.
- Rest for 5 hours.
- At the end of discharge, update status will change to 06

Extract golden file for mass production.

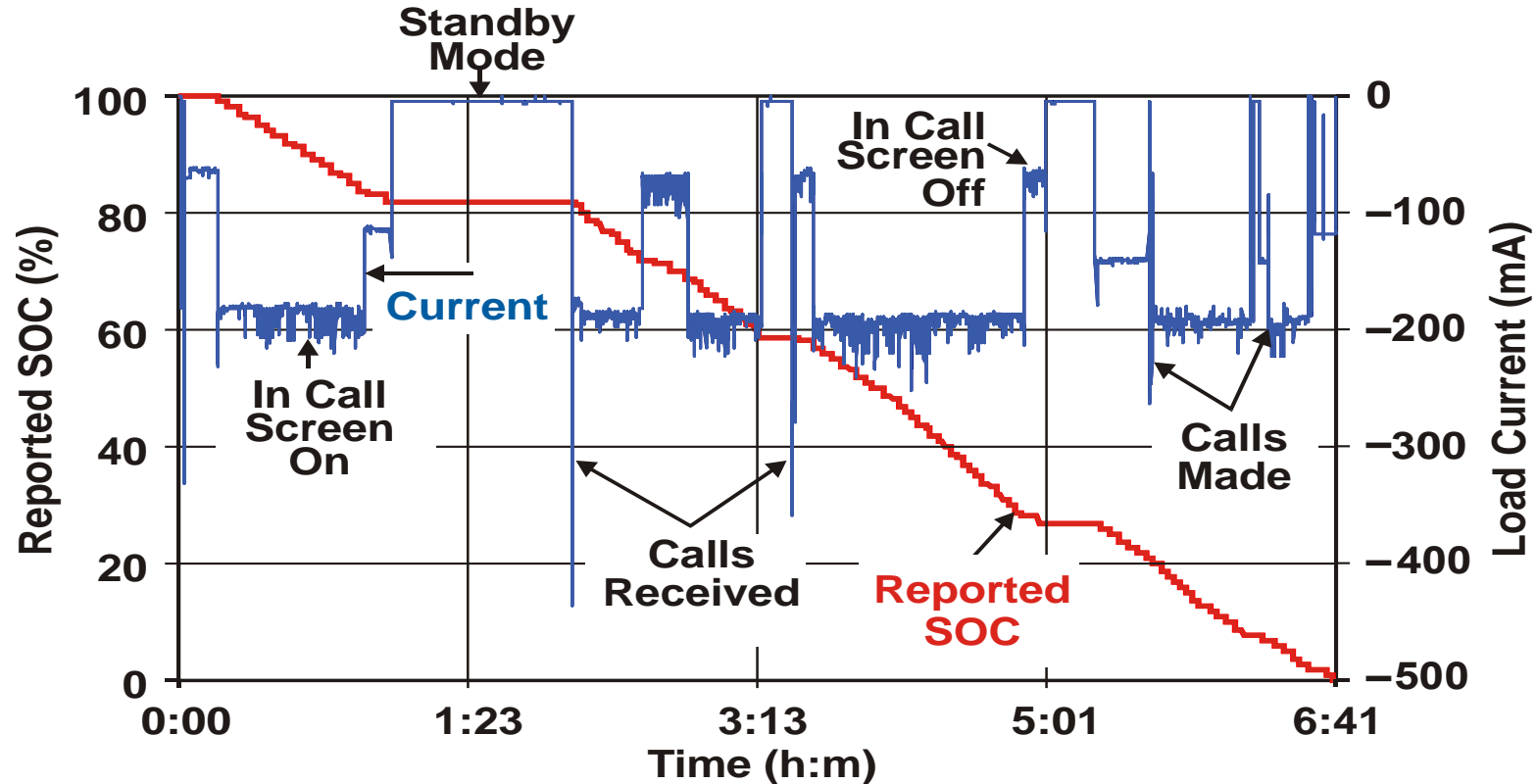
- Extract a golden srec, bqfs or dffs. You can calibrate 10 boards and program the average of the calibration parameters in your golden file .

Evaluation tests

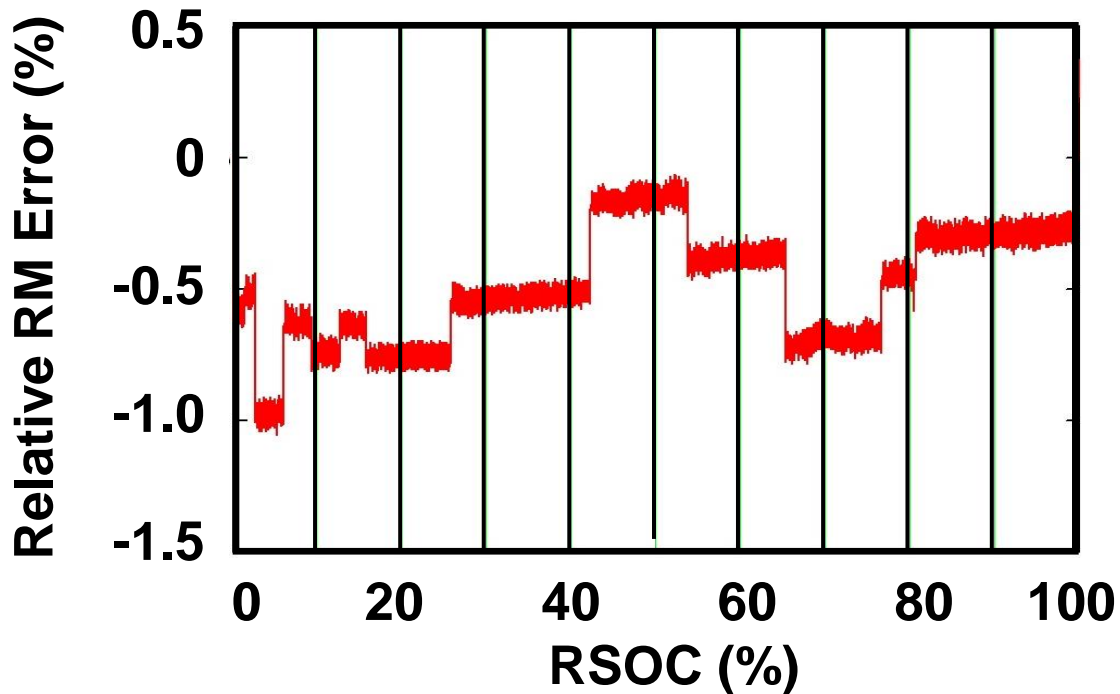
- You can set up an the actual load to perform evaluation tests as below



GSM phone test: reported SOC and current



Fuel-gauge test results with a smart phone



- Up to 99% accuracy at room temperature

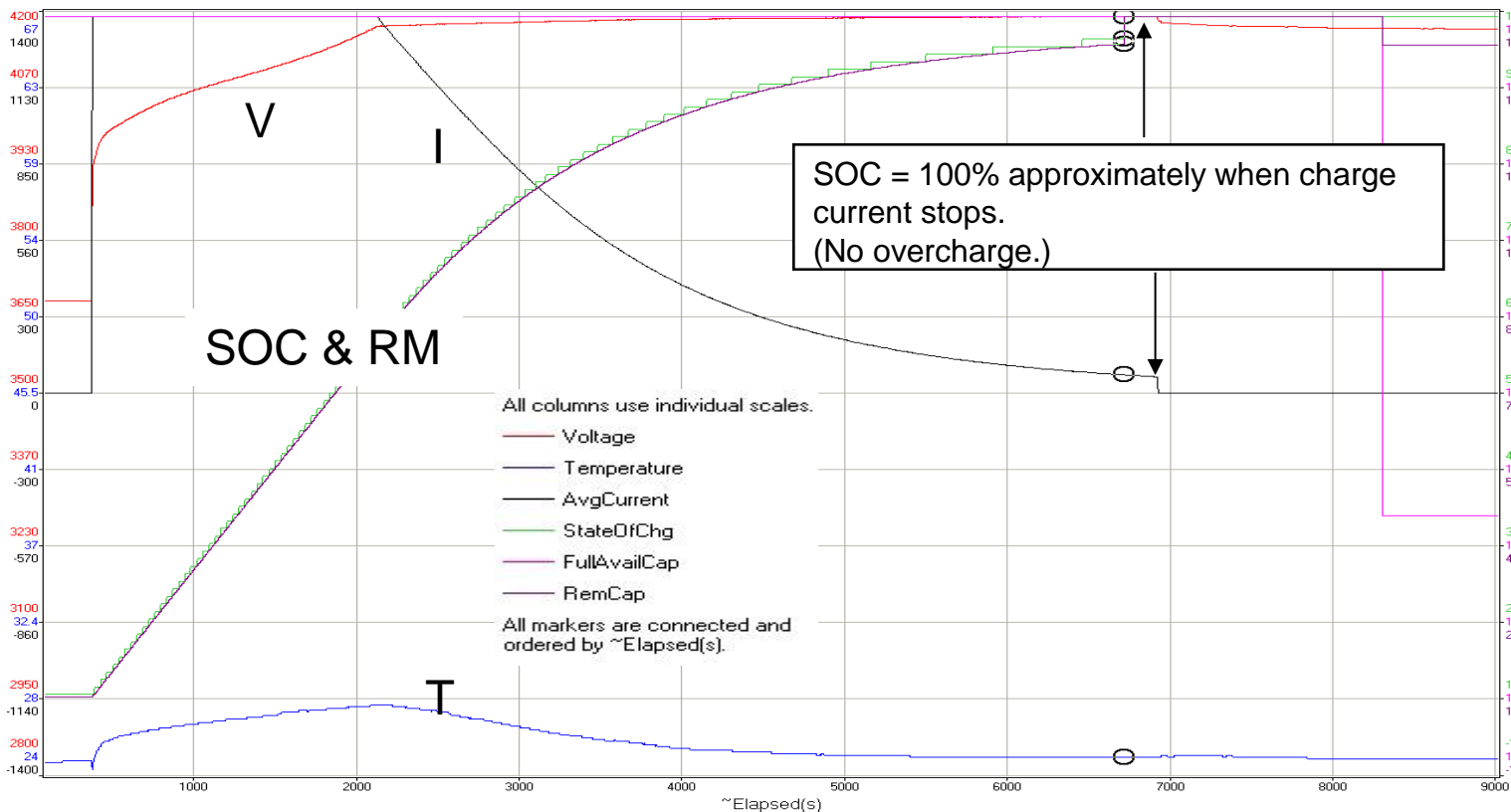
Accuracy Evaluation

- Simple inspection
- Detailed error calculations
- Common sources of error

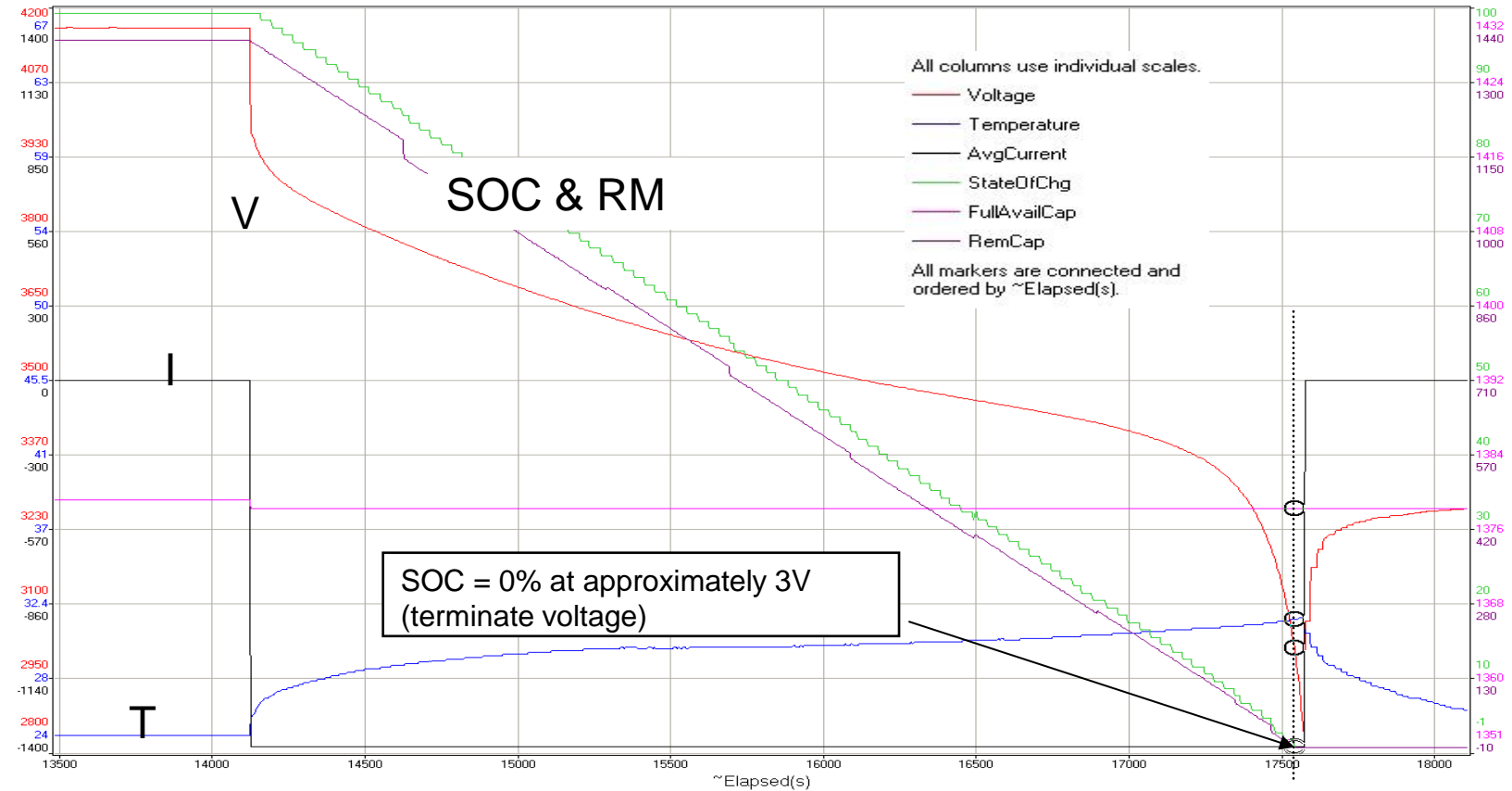
Simple inspection

- Log gauge data every 1-10 seconds
- Import to spreadsheet and plot voltage, SOC, current, etc.
- Inspect
 - Does it report 0% at or near your true terminate voltage?
 - If not, by how much capacity and/or run-time was it off?
 - Does it look smooth or have large jumps?
- Very subjective and difficult to gauge relative error magnitude

Example: inspect a 25°C charge at 1C rate



Example: inspect a 25°C discharge at 1C rate



Detailed accuracy calculations

- Log gauge data every 1-10 seconds & import to spreadsheet
- Calculate a new column: $Q_integrated = \text{rolling sum of current_reading} * \text{time_since_last_log_point}$
 - Start from full condition
- Calculate $FCC_true = \text{Integrated capacity from fully charged state down to termination voltage}$
 - Simple integration: $\text{capacity} = \text{constant current} * \text{total discharge time from full to terminate voltage}$
 - If using varying load, you need to calculate each log point $\text{Current} * T_{\Delta}$ and keep a rolling sum.
- Calculate a new column: $Qrem_true = FCC_true - Q_integrated$
 - True remaining capacity at any point (mAh)
- Calculate a new column: $RSOC_true = Qrem_true / FCC_true$
 - True relative state of charge (%)
- Calculate a new column: $Qrem_error = (Qrem_true - Qrem_gauge) / FCC_true$
 - $Qrem_gauge$ is “Remaining Capacity” in IT gauge log files
- Calculate a new column: $RSOC_error = (RSOC_true - RSOC_gauge)$
 - $RSOC_true$ is “State of Charge” in IT gauge log files
- Plot voltage, $Qrem_true$, $Qrem_gauge$, $Qrem_error$
- Plot voltage, $RSOC_true$, $RSOC_gauge$, $RSOC_error$
- Summarize in table showing
 - % error at the point of 15% true SOC
 - % error at the point when termination voltage is reached
 - Max % error

Extreme RSOC analysis: 1C discharge at 0°C

Tricky corner case:

- High discharge rate at low temperature

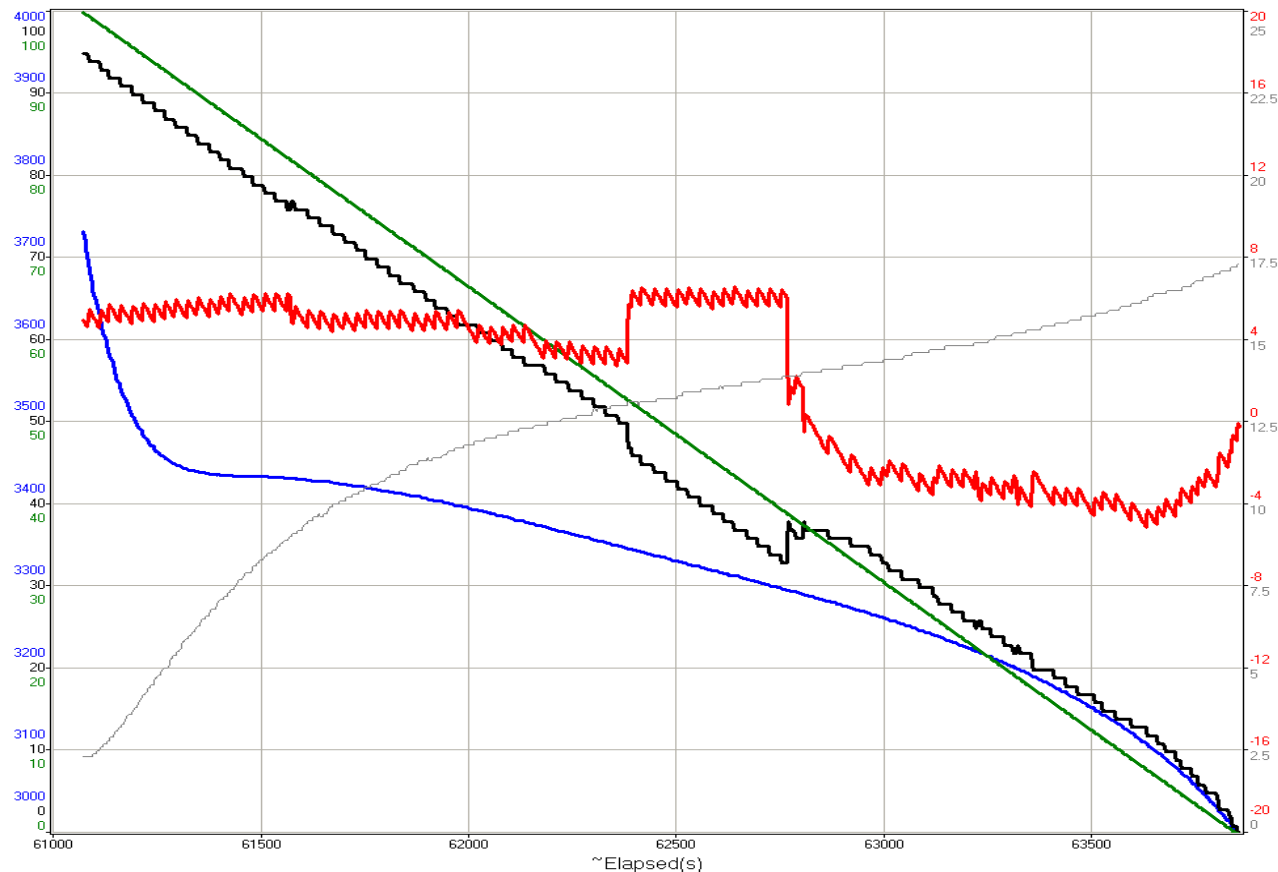
Results:

- RSOC starts with some underestimation then slightly overestimates but converges to 0% at true termination voltage

All columns use individual scales.

— Voltage
— StateOfChg
— RSOCtrue (1)
— RSOCError% (1)
— Temperature

All markers are connected and ordered by ~Elapsed(s).



Extreme RemCap analysis: 1C discharge at 0°C

Tricky corner case:

- High discharge rate at low temperature

Results:

- Remaining Capacity starts with some underestimation then slightly overestimates but converges to 0% at true termination voltage

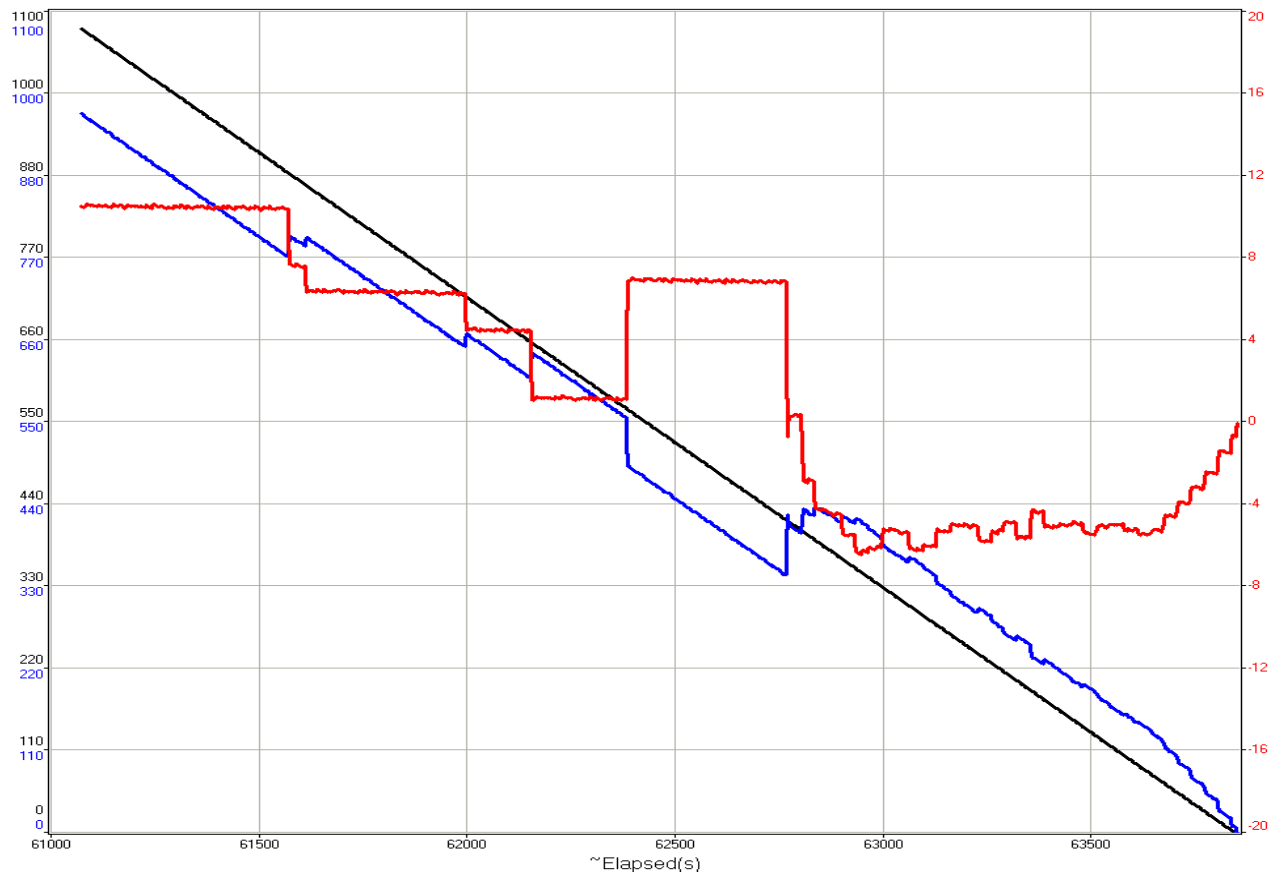
All columns use individual scales.

— RMtrue

— RemCap

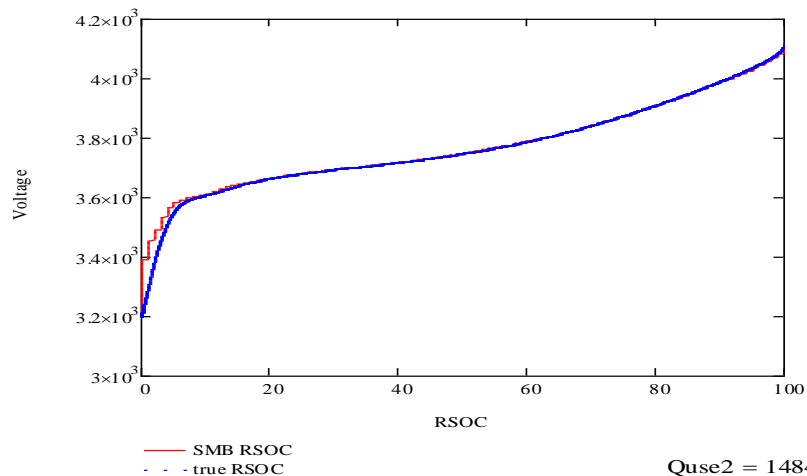
— RError%

All markers are connected and ordered by ~Elapsed(s).



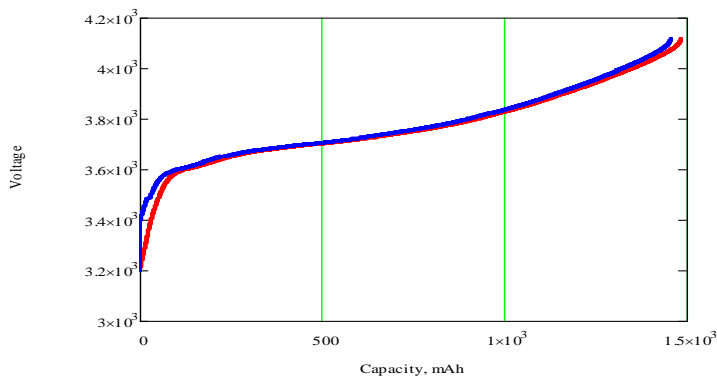
Example Mathcad analysis

True vs reported RSOC

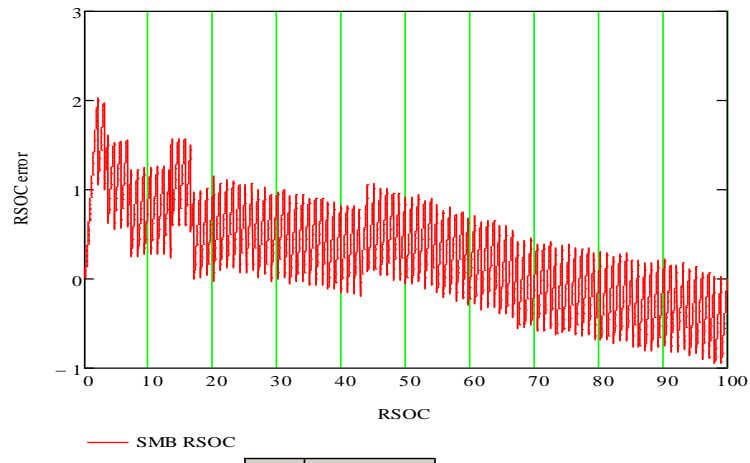


Quse2 = 1484.9975

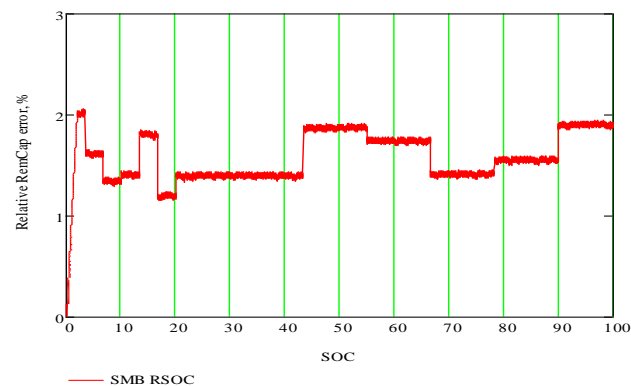
Remaining capacity



RSOC accuracy



Relative RemCap error



Common sources of error

- Wrong or skipped calibration
- ChemID or modeling parameters not correct for actual cell in use
- Golden file not optimized: skipped or bad learning cycle leads to wrong initial Qmax and resistance values
- Temperature error: make sure thermistor is actually attached to the cell during test
 - It needs to know the cell temperature, not the ambient temp.
 - Cells typically self-heat during discharge.
- Gauge not allowed to accurately measure starting capacity
 - Ex. Attach unrelaxed cell so gauge uses wrong starting voltage to calculate initial SOC
 - Ex. Charge or load is present when gauge is initialized or cell is attached
- Thermal modeling:
 - Bare cell at low temp will not heat up much
 - Typical system will be enclosed and have more self-heating
 - Does your gauge handle either situation or does it need configuration of thermal modeling parameters?
 - Symptom: large temperature changes during test may result in inaccuracies and/or capacity jumps unless gauge is configured to handle it
- Selecting wrong LoadMode (constant current or power) and LoadSelect for application
- Avg I/P Last Run in dataflash not initialized to expected rate to be used in test
- Overcharging or undercharging cell by charging at high or low temperatures
- Significantly overcharging by tapering to a lower current than gauge is configured to use for full charge detection
- Using old firmware version: TI is always continuously improving the IT algorithm to better handle corner cases or odd usage conditions!

Conclusion

- The key considerations when designing a single cell battery gauge system was analyzed.
- Gauge performance is closely associated with the algorithm implemented and the correct configuration.
- This design walk through can be expanded to work with any single cell gauge application

Questions?