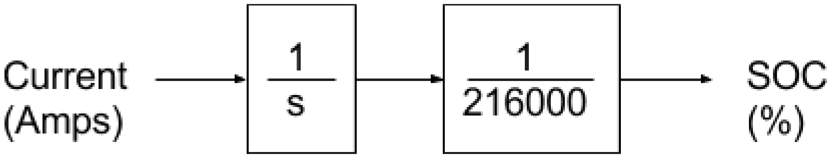
To: LFEVY42016 Team
From: Geoff Nudge
Date: 5/7/2016
Subject: QAR001a - Charge Algorithm

Abstract

It is a requirement in the SOW that a charging algorithm is implemented in software to facilitate charging of an accumulator, and to provide a reasonable state of charge estimate. While the SOW states that a cell model and coulomb counting are to be included in this algorithm, coulomb counting was used alone due to time constraints. It was found through testing that an accuracy to within 8% SOC was achievable with this method. Charging is performed by connecting the TDK Lambda charge to the Anderson port with a cable that has a jumper across it male signal pins (photo attached).

Technical Findings

Coulomb counting is a method of tracking the stored charge in an accumulator. This is accomplished by simply integrating the pack current over time, and using appropriate factors to translate coulombs into a state of charge percentage.



A 60 Ahr cell can hold 60Ahr*60min/hr*60sec/min = 216000 Asec (coulombs). This is an open loop system and is marginally stable (unstable for practical purposes). Any error in the measurements of time or current will accumulate over the course of a charge or discharge cycle.

The data sheet of the LiFEPO4 cells used (http://www.batteryspace.com/prod-specs/6334.pdf) shows that a cell may be considered fully charged at 3.7V with charging current of 20 A applied. It also shows that 2.7 V with a draw of 180 A may be considered dead. To mitigate the drift in SOC as a result of integrated errors, the software always transitions to fully charge or discharge as a result of a cell reading outside this range.

The coulomb counting method was achieved in software by a task in the Atom RTOS that would compare the current system time (in 10^-2 sec) to the system time of the last integration, and multiply this by the available current measurement. The total coulombs count up continuously and stop counting down near 0. SOC however peaks at 98% until a 3.7 V reading is made, and bottoms out at 1% until a 2.7 V reading is made.

A simple cell balancing was also implemented. If at anytime during charging, there is a gap in cell voltages of greater than 25 mV, the cells with voltages above this gap are set into AMS bypass mode. Because this was only observed in the last 2 to 3 min of charging and with a duty cycle of about 50%, there was no significant change in heat sink temperature.

On boot up, the PacMan computer assumes a value of 50% SOC. This allows a user to charge or discharge the pack after assembly as needed. If either limit (3.7V or 2.7 V) is read at any time (regardless of SOC), the SOC and counted coulombs are updated, and the pack is considered charged or dead as appropriate.

After testing with 4 complete charge discharge cycles, the desired behavior has been observed. It is typical for a charge or discharge operation to continue past the expected time to reach 100% or 0%. The largest deviation from expected time was 15 min 40 sec during a charge which would result in 108% SOC if unchecked (some of this can be accounted for by the cells bypassing). The read out was held at 98% during this time until a 3.7 V cell reading was made. Relevant data in csv and xlsx (with graphs) formats is available at the following address:

https://sites.lafayette.edu/ece492-sp16/files/2016/05/AcceptanceTestData_TSV.zip

Recommendations and Conclusions

It was observed that cell temperature affected cell voltage and therefore SOC. Also errors in SOC were not consistent (as one would expect), and ranged from near 0 to 8%. As a result, a future iteration should consider conducting more charge and discharge cycles, varying load and temperature, to characterize the accumulator as a whole. The developed accumulator model could be used to create a feedback loop in the SOC calculation that will provide a consistent and accurate reading.

While these improvements could be made, it should also be noted that the charging algorithm as it is now implemented consistently reads 1% and 98% for some period of time (during discharge and charge respectively), and never a value outside of the range of 0 to 100. The algorithm provides an output a user may rely upon to determine if the accumulator is in need of a charge, or has sufficient charge for a given task.

If changes are made to the cell balancing algorithm, it should be noted that continuous bypass will result in heat sink temperatures in excess of 100 degrees C after approximately 5 min.

ATP list the following for QA regarding charging algorithm:

<u>QAR001a</u> - Charge Algorithm

1. Mathematical analysis of battery charging/discharging state of charge. The method includes a cell model and coulomb counting.
2. Testing on accumulator test stand.

    a. charging starts appropriately, normal operation
    b. charging stops appropriately, normal operation
    c. charging stops appropriately, all failure modes
3. Charging a discharged TSV accumulator with LiFePO4 cells

All of these items were successfully completed. The mathematical analysis is included above. Testing in the test stand was performed successfully, but no data was collected. Data was collected for charging and discharging with the cells, and is available at the link provided above.

```
STATE:RDY    SOC:095%
SAFETY LOOP: OPEN
V:23.3 A: 000.056
T>v0.14 002:02:22:28
```

The port labeled CHRG/LCO is the Anderson port used for charging and low current output. It's female signal pins should be jumped to indicate charging. A charge cable should have its male pins jumped to accomplish this.

Also, the black and red power pins have been modified to allow mating with the finger proof versions used in the existing charging cable. Additional packs should include the finger proof version if this modification is not desireable.