

Open Source Multidisciplinary Load Bank

Abstract

Introduction

In their 2010 National Energy Policy Recommendations, IEEE USA identifies energy storage as a critical technology for electric vehicles, the Smart Grid, and for renewable energy sources like wind and solar. Many undergraduate and graduate students are interested in battery electric vehicles, plug-in hybrids, and renewable power systems that utilize battery energy storage systems.

The electric drive train has proven itself to be extremely reliable and efficient. However the electric vehicle must be economically viable in addition to clean and efficient. Typically Li-ion cells are used for energy storage because of their high energy densities compared to other battery chemistries. For acceptable electric driving ranges this requires very large battery packs that represent a very significant percentage of the total vehicle cost. Because of the packs high cost its life time substantially affects the affordability of the vehicle.

Cycle life, energy density, and cost are key factors in the successful application of battery energy storage. In many applications, the aspect of cycle life has not been addressed as thoroughly as performance or cost. One driving reason for this is that life cycle tests can require months or years to perform at room temperature and require very expensive load banks, chargers and data acquisition systems. The cost of these systems is prohibitive for graduate studies unless significant funding is available from a grant or existing assets can be used. The situation is even worse for undergraduates who want to study energy storage, as typically even less funding is available to support them, regardless of their talent, enthusiasm, and dedication.

We believe the lack of affordable electronic load banks, chargers, data acquisition systems, and software to run these systems is a barrier to rapid progress in energy storage systems. Therefore in 2009 we began a project to develop an affordable open source, open hardware system for performing life cycle measurements on energy storage systems including batteries and ultracapacitors. This paper describes the system design philosophy, design choices, the initially targeted load cycle, and the integration of students into the development of the open source system. It is hoped that this will lead to further discussions and exchange of ideas about how to develop and apply open hardware in a variety of educational settings.

A need for test equipment

Possible Solutions: Commercial Equipment

A Choice: Our Solution: The battery Cyclor

In response to the difficulty of obtaining the capital equipment necessary to conduct battery testing, we have created the battery cyclor project. A solution that integrates all four elements of testing similar to a commercial solution; data acquisition, cell load and charge control, environmental control, and host PC management software. It is a complete system implemented in dedicated but open hardware that is vastly scalable to meet the specific type and size of testing a user plans on conducting.

The scalability of our system means it is capable of testing full size and readily available production cells, as well as smaller coin sized test cells.

Unfortunately, open hardware does not mean free hardware. None the less, battery testing equipment is a market where volume is small and margins are high, there are great savings to be had by building an open solution like the battery cyclor. We estimate that we can cut the cost per watt of testing equipment to nearly a tenth of many high end commercial systems.



Design Philosophy: Open Hardware

Battery development attracts a variety of backgrounds, many focused in material sciences or chemical engineering. Thus employing free and easy to learn development tools is essential to make sure that collaborators are able to take part. The assumption that users will already have access to and experience with more conventional tools cannot be made.

One of the most important elements of the project has been to keep the bar to entry as low as possible. By using freely available development tools such as Arduino, Eclipse, and Eagle we are able to ensure that everyone that would like to participate in the project will have the ability to.

Arduino

Arduino has set the standard for open hardware, taking the success of open software and bringing it to the hardware development world. Arduino's success stems from building on the wiring and processing platforms that use abstracted C and Java programming, allowing people who do not necessarily understand how computers and microcontrollers work to use them. We look to emulate this model of an open platform in the battery cyclor, thus building on the Arduino platform for the hardware of the battery cyclor was a natural choice.

JAVA + Eclipse

The Host PC software of the Battery Cyclor is built on JAVA, which is an excellent software platform for open source projects. It is available free under the GNU General Public License, which keeps it free (with a few exceptions in the core of the framework of the JVM). JAVA is heavily standardized and extremely

well documented. SUN specifies a standard for code comments which makes JAVA source code easy to understand, implement, and compiles web-page-like document known as JAVADOCS.

There are many free and open development environments that support JAVA. Eclipse is one of the most feature rich and popular, and it supports JAVA development as well as many other languages including C/C++ and web-applications. Eclipse supports many plug-ins which make development for many different platforms easy. Enabling SVN support through plug-ins allows for check-ins and check-outs directly through the program's interface. Eclipse supports advanced coding features such as auto-formatting and executable generation. Both the environment and plug-ins are very well documented.

Google Code

In a collaborative environment having a good repository is just as essential as having good development software. Google runs a free project hosting service for all Open-Sourced projects, Google Code. For an open platform to be truly successful it needs to allow its users to make improvements and share those improvements with the rest of the open source community. Working within a Google code repository will help us achieve this

The hosting service provides revision control supporting both Subversion and Mercurial, revision control is a crucial tool to provide the foundation laid to the masses of people looking to implement the open source hardware and software. This service makes collaboration within the team convenient, but also will make collaboration on a larger scale possible. The website also provides a bug-tracker system, a wiki for documentation and a file-download feature for distribution of released software and hardware.

Eagle

The battery cyclers relies to a great extent on hardware external to the arduino development board. The system also requires load and charge banks to manipulate the cell. Developing the hardware to do this requires schematic entry and board layout capability. Eagle by cadsoft provides a professional grade solution free for nonprofit use. Supporting eagle allows us to publish schematics and board design in a format that other contributors can easily access and manipulate for free.

Good Documentation

A great amount of time has been spent documenting the architecture of the system and the layout for software development, this has been essential to fostering a successful collaborative project and paving the way for the work that needed to be done. Integrating this into a wiki within the provided Google code structure allows us to make the documentation part of the collaborative process.

Related Design Decisions: System Details

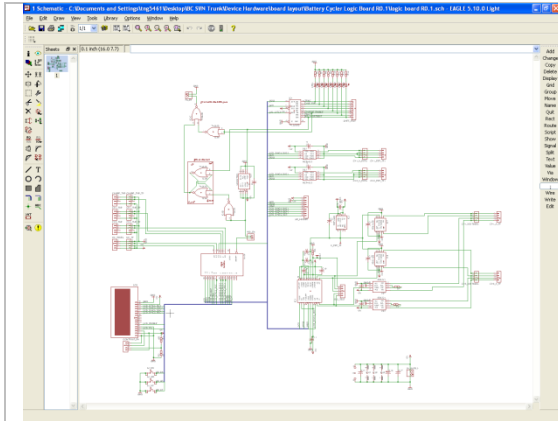


Figure 1



Figure 2

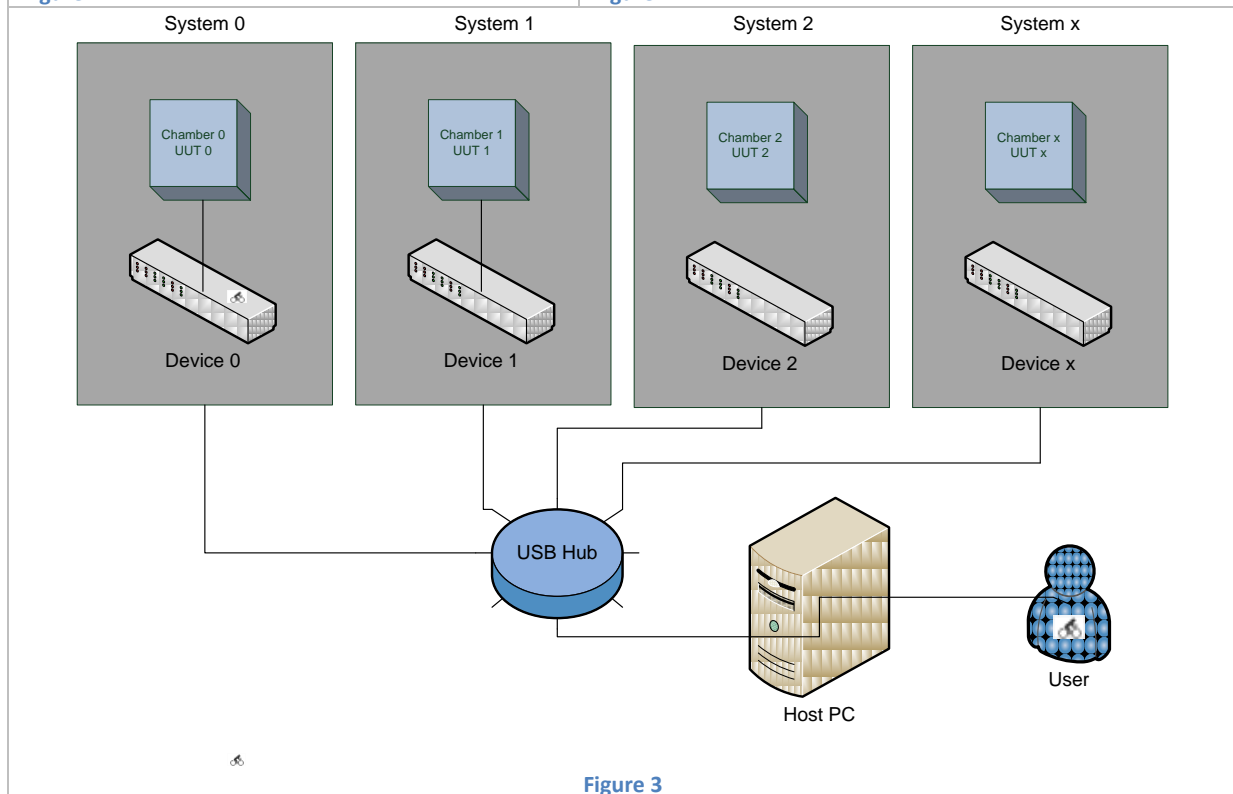


Figure 3

System Topology

The battery cycler system is comprised of

Channel Scalability

Many commercial testing solutions are scalable, however only within the limits of their base systems power supply and controls. Often when only small scale testing is initially desired the cost of the base system alone is prohibitive.

The battery cycler fully integrates the charging power supply and control within each device. Each device only needs USB connectivity to a host PC for control. Thus multiple devices can be connected through a

USB hub. The only limit to expandability with a single Host PC is the number of interrupts and memory space within the PC, figure 3 illustrates this.

Power Scalability

Similar to cell count battery experimentation occurs at a number of different power levels. While the initial intentions of the battery cycler platform are to focus on common production cells in the 2Ah to 5Ah range, it is scalable above and below this. Each cell channel is constructed of a number of current sources in parallel that make up that channel's current handling capacity. Having an open system allows a user to precisely define the size of a channel depending on their resolution and power needs. In addition, multiple channels within a single system can be combined to increase power for a single test.

The table below shows the many power configurations available within the current system architecture.

Environmental Control

Life cycle testing can often take months to years to perform at ambient temperatures. Because of this cells are often environmentally stressed during testing to accelerate degradation mechanisms.

Environmental control is a core function of the battery cycler providing an integrated PID loop, temperature feedback, and triac to switch heating elements and provide temperature regulation.

Environmental chambers can be easily constructed and tuned to suit the needs of a user. Figure 2 shows an example miniature thermal chamber with a 100W heating element, circulation fan, and ceramic containment pot for units under test.

Safety

Safety is paramount when considering testing that will be pushing high energy density cells to failure. An exhaustive failure mode analysis shows the amount of work that went into ensuring that testing that is performed within the battery cycler platform is fail safe. A robust communication protocol, as well as hardware watchdog, cell fusing, temperature monitoring, and software definable electrical and thermal limits insure that a system fault will always protect a cell before causing a failure.

Data acquisition

Data is reported and logged at 10Hz providing measurements of cell voltage, current, temperature, cumulative energy in and out, as well as a state of charge and test progress indicators. This raw data can be post processed to determine important cell characteristics such as ESR.

Host Software

- Test organization

- Test results

Supported Load Profiles

The system provides the capability to stress and characterize critical parameters of cells. To aid this effort and due to the system being developed with testing for hybrid electric vehicles in mind, the system is designed to implement standardized testing procedures such as those defined in the DOE Battery Test Manual for plug-in hybrid electric vehicles.

Standardized charge depleting, charge sustaining, and characterization test are built into the devices firmware. Performing calendar life testing following these standards is merely a matter of defining scale factors, safety limits, and duration.

In addition we provide an interface for defining cell load profiles and test sequences by the user. The system is designed to operate in a variety of modes that can load the cell maintaining a constant current, voltage or power with definable limits.

Integration of Students: Integrating Engineering Education

As an integrated solution by way of a multidisciplinary project, the Battery Cycler has allowed the students that need the hardware to participate in the development of it.

This process began back in 2009 as a proposal for a summer undergraduate research project developing the foundations what is now the Battery Cycler project. This was done with the following goals in mind:

- Cost under \$200/test configuration
- Independent control of cell charge discharge rate
- Control of cell environmental conditions
- Ability to monitor cell response to loading and log cell health related data.
- Redundancy in case of power failure

It was immediately clear that the project needed to be broken into several major components, Device software development, Host PC software development, and device hardware development.

During the course of the summer three students joined the project, a graduate Software Engineering student joined and formed the initial framework for the host PC software discovering many of the major challenges that would be encountered communicating without hardware. In addition two electrical engineering technology undergraduate students began work developing the load and charge banks that would interact with test cells, as well as defining the external hardware needed for data acquisition, output control, and safety considerations.

A great deal of time was spent defining constraints for the many different sub systems. Learning how to develop an efficient and robust communication protocol as well as dealing with many of the headaches of building a system that is robust enough to run for months without crashing.

The battery cycler is a diverse project that provides not only an excellent platform for the testing and education of battery technologies, but many other engineering disciplines as well. An educational environment can benefit from the project as it offers many classic engineering problems in an applied setting that can tie in with an engineering curriculum.

For example the environmental control provides a classic example for tuning the PID control for different chamber and heating element designs, an excellent problem for a Control systems class. The load and charge banks are ideal for an electronics class exploring power supply design, and loop analysis

contributing to the systems stability, total bandwidth, and optimizing to meet the systems needs. The embedded environment provides a project for basic microcontroller programming and studying the systems architecture. The host PC software is built in java, a language that reaches across many disciplines in the sciences.

While students move on, the project is refreshed with new students from different programs that are interested in participating as independent studies and paid graduate work. The project continues to include students with a broad range of interests and experience, all willing to contribute and learn.

The Future

We have drawn the framework for a powerful multidisciplinary project that provides a solution to a choke point in battery research for educators.

We have grown the project to revision one, a point where it meets our original goals and we can begin the testing that originally inspired the project. We are extremely excited to have reached these goals and be able to release our work into the wild.

With the amount of work that was required to bring the battery cyclers to revision one, it is only natural to be confronted with countless ideas for improvements and additions to the system along the way. We are excited to see the platform continue to grow with new innovations and collaboration with other students and institutions.