A Lithium Polymer Battery State of Health Assessor, Version 1.4

By R. G. Sparber

Purpose

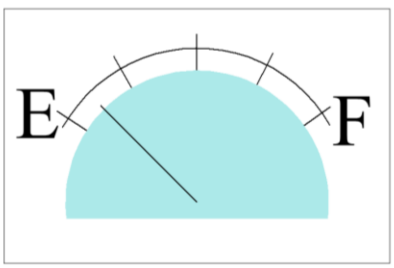
Charging and discharging a Lithium Polymer (LiPo) battery causes it to degrade over time. By measuring its ability to source current over time (ampere-hours), it is possible to track this degradation. The information is useful when calculating its ability to store energy and knowing when the battery is at the end of its life. See Appendix I for an overview of how these batteries behave.

User's Guide

1. Start up the laptop and bring up TeraTerm
2. Within 15 seconds, connect the Load Tester via the USB cable
3. Follow the instructions on the screen to run the test
4. When the test completes in about 7 hours, the final line will be the total number of ampere-hours drawn from the battery.
5. Close the log file and open it in Excel to graph results.

Contents

State Of Charge And State Of Health



Picture a gas tank on an internal combustion engine car. It holds 10 gallons of gas. I have a gas gauge that registers Full when 10 gallons are present, ½ when 5 gallons are in there, and Empty when only fumes remain. The gas gauge reads out in fractions and not gallons.

An ideal battery could tell the same story. Rather than talking about gallons, there would be ampere-hours[2]. When the battery is fully charged, it could hold 10 ampere-hours. When at ½, 5 ampere-hours. And when Empty, no more current would flow. My battery charge gauge reads out in fractions and not ampere-hours. When I see that my eBike battery is at ½, I know that I better be heading back home[3]. The charge gauge is telling me the battery's State Of Charge (SOC).

[2] This is not a perfect comparison but bear with me.
[3] The nearest gas station can supply me with a snack, a drink, or a bathroom, but it won't recharge my battery.

As the gas tank ages, its capacity doesn't change. But as a battery ages, it stores fewer ampere-hours. A new battery would have a State of Health (SOH) of 100%. After each charge/discharge cycle, it loses a tiny bit of this capacity. After hundreds of cycles, the SOH has dropped low enough that we must replace it.

SOH and SOC are linked. I look to a battery's SOH to tell me how much capacity it how contains. SOC tells me how much of that capacity I have left.

Much has been written about determining the State of Charge of a battery. I use a "Coulomb Counter" for this task. It directly measures the number of ampere-hours flowing from the battery. I reset it to "Full" when I charge the battery. It tells me what fraction of this capacity is left before arriving at "Empty."

My Application

The Lithium Polymer (LiPo) battery in my Lectric XP eBike is a monster. When fully charged, it outputs over 50 volts at up to 20 amperes. I recharge it after each ride. My range is a function of many factors, including terrain, my weight, the wind, tire pressure, speed, and, of course, the SOH of the battery. With all of these variables, it is difficult to know when the battery has degraded enough to warrant redefining "Empty" in my Coulomb Counter.

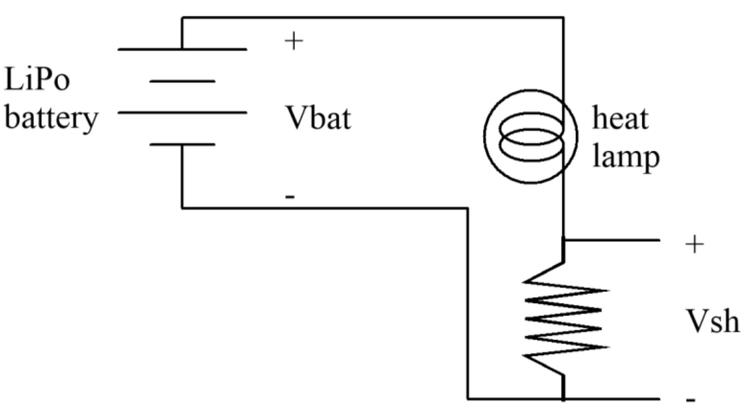
A State Of Health Assessor

To address this missing piece of the puzzle, I developed this SOH Assessor (SOHA). It draws an almost constant current while recording the LiPo's output voltage and current. This data is used to calculate a few interesting bits of insight. Of primary importance, I get the total number of ampere-hours in a fully charged battery. This is done by reading the current once per minute. Assuming this current is constant over this minute, I calculate the ampere-minutes consumed. I convert to ampere-hours by dividing by 60. When I sum all of these ampere-hours over the run of the test, I get the total number of ampere-hours sourced by the battery.

Running this test every four months allows me to monitor the total ampere-hours under the same test conditions to see the SOH.

Looking at output voltage versus time is interesting, but I don't see any long-term value in it. The least helpful information is SOC versus battery voltage. This graph is only valid at this particular current drain and this point in the battery's life. Yet, many people treat it as gospel.

The SOHA comes in two flavors. It can be simple but require the user to take many readings with a voltmeter, or it can be more complex with an Arduino collecting and outputting the data. I did one run with the voltmeter and promptly started work on the Arduino solution.



The heart of the SOH device is a 125VAC 250-watt heat lamp. It has a few great features:

- low cost
- readily available
- Visual indicator

Home Depot sells these heat lamps for $10, and you get two bulbs in the package. When connected across my LiPo, the current is around 1.5 amperes which is close to my normal drain. I can buy a 75-watt resistor for around $10, but it needs a heatsink which adds cost and complexity. I also have to pay for shipping and handling because it is not available at any local stores.

The heat lamp has a dull glow, but it is easy to tell current is flowing.

I measure current by having a 0.5-ohm 5-watt resistor in series with the heat lamp. The voltage across this resistor, V_{sh}, divided by 0.5 ohms, is the current.

The results are valid because the test conditions are consistent for each run. At this time, I expect that looking at the slow degradation of ampere-hours will be the most valuable insight.

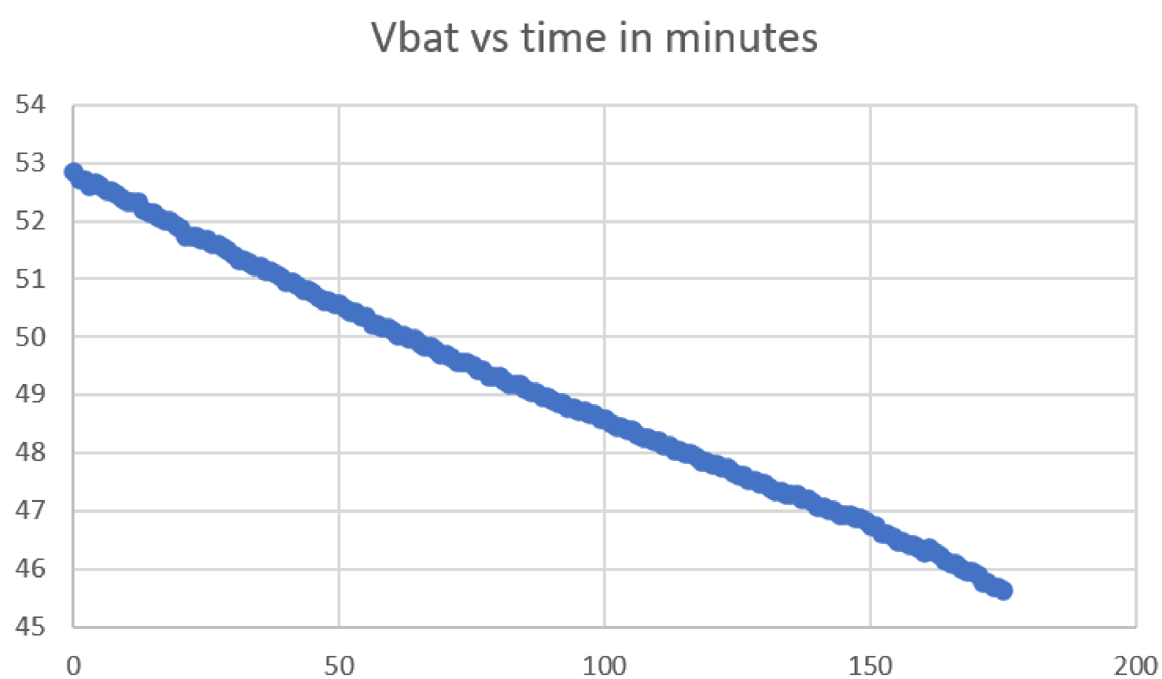
If your application requires a different current, screw in another bulb. I did an experiment using six 12V 50 watt halogen bulbs in series. The problem was that when cold, they drew 20 amperes. I was afraid that would damage the battery. Once hot, the resistance went up, and the string drew around 3.5 amperes. The heat lamp has far less drama.

SOH of a Defective Battery

I had a battery that suddenly decreased the range I could get on my eBike so I wanted to measure its SOH. When new, it should have had a capacity of 10.4 ampere-hours. I connected up my heat lamp and shunt resistor and proceeded to measure $V_{ba}t$ and V_{sh} every 15 minutes until the battery turned itself off. It was a crude test yet gave me a clear idea of the battery's SOH. I measured 4 ampere-hours. I also found that the battery shut down at 45 volts rather than the expected 40 volts.

This experience taught me that the SOH test was helpful ***and*** that I didn't want to do another manual test.

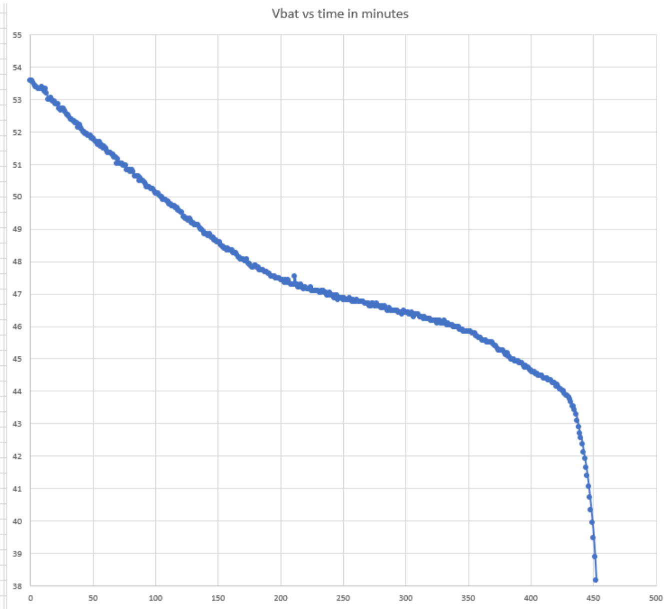
An Arduino compatible Pro Micro solved my problem. With a bit of software, it took a reading every minute and dumped it to my laptop's screen. After a few keystrokes, this data was in an Excel spreadsheet which produced the various reports.



The battery voltage started at 53V. It should have been closer to 54.6V. It fell almost linearly and shuts down around 45.5V instead of 40V. Without any comparison, this graph didn't mean much. However, seeing a capacity of only 4.04 ampere-hours clearly showed the battery was unwell.

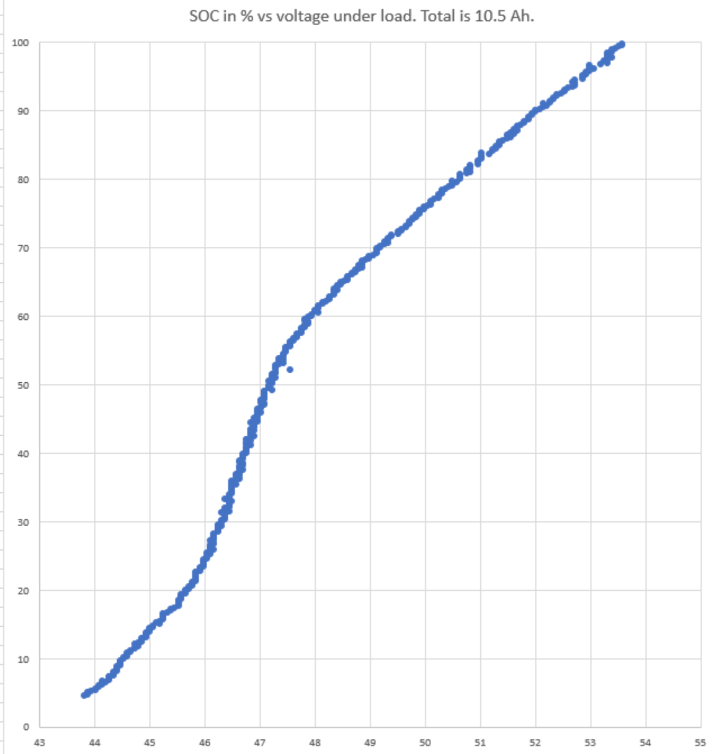
Armed with this data, I was able to convince Lectric that the battery was defective. Lectric has excellent customer service, and they traded my broken battery, which was under warranty, for a new one.

SOH of My New Battery



My new battery had a vastly different profile. The capacity was 10.45 ampere-hours which is a little more than rated. The graph of voltage versus time showed a more complex drop in voltage, plus shutdown was at 38 volts.

Looking at the cumulative ampere-hours, I could see that the amount of energy extracted between 44V and shutdown was only 0.06 ampere-hours. Yet, at 38V, the average volts per cell was 2.9, close to deep discharge. That can damage the cell. I decided to add hardware and software to end the test at 40V, 3.1V per cell.



Although of limited value, here is a plot of SOC versus battery voltage. It is only valid for a load current of about 1.5 amperes and this particular new battery.

Voltage	SOC, %
53.5	100
52	90
50.6	80
49.1	70
47.9	60
47.2	50
46.7	40
46.3	30
45.7	20
44.5	10

The Arduino Based SOHA

The Arduino enables me to add a few nice features. When I first turn the battery on, the system measures the open circuit battery voltage every 10 seconds for a minute. This gives the battery time to settle down and show the open-circuit voltage[4]. Then I connect the load and measure the battery voltage and current every minute until the voltage drops below 40V. The load is then removed to prevent deep discharge.

While the circuit waits for the battery to be turned on, a red LED flashes on for 1 second and off for 1 second. As long as the battery voltage is greater than or equal to 40V, the LED flashes 0.2 seconds on and 0.2 seconds off.

Each reading is sent to a terminal emulator running on my laptop with the format

Time, Vbat, cumulative ampere-hours

where Time is in minutes since the load was connected and V_{bat} is the battery voltage. Each minute the program reads the shunt voltage and calulates the load current. It then calculates the ampere-hours over the last minute. The cumulative ampere-hours is then output. When V_{bat} falls below 40V, the test is over.

This data format is compatible with Excel as a Comma Separated Volume (CSV).

See Appendix II for more on the software. Appendix III contains details of the hardware.

[4] By knowing the open circuit voltage, voltage under load, and the load current, I can calculate the Equivalent Series Resistance (ESR). Some claim that this is an indicator of SOH.

I welcome your comments and questions.

If you want me to contact you each time I publish an article, email me with "Subscribe" in the subject line. In the body of the email, please tell me if you are interested in metalworking, software plus electronics, kayaking, and/or the Lectric XP eBike so I can put you on the right distribution list.

If you are on a list and have had enough, email me "Unsubscribe" in the subject line. No hard feelings.

Rick Sparber
Rgsparber.ha@gmail.com
Rick.Sparber.org

Appendix I: A Little Battery Theory

LiPo batteries are commonly referenced by two electrical parameters: voltage and capacity. A single LiPo battery cell reads about 4.2 volts when fully charged and is damaged if discharged below about 3 volts. It is common to say that one cell outputs 3.7 volts. This is the "name" or "nominal" value. My battery is built from 13 of these cells in series, so its nominal value is

$$13\ cells \times \frac{3.7\ volts}{cell} = 48.1\ volts.$$

When fully charged, it is around 54.6 volts, and when fully discharged, I can expect 39 volts.

Capacity is measured in ampere-hours. If I draw 1 ampere from the battery for 1 hour, I extracted

$$1\ ampere\ \times 1\ hour = 1\ ampere - hour.$$

But I can also pull ½ ampere for 2 hours and have

$$\frac{1}{2}\ ampere\ \times 2\ hour = 1\ ampere - hour.$$

Ideally, any combination of current and time that equals the same ampere-hours depletes the battery by the same amount.

My battery is rated at 10.4 ampere-hours (Ah). When new, I can pull almost[5] any combination of current and time that equals 10.4 Ah. After repeated discharge and charge cycles, this capacity degrades. Its "State Of Health" (SOH) goes from 100% to a smaller number. It is also possible for one or more cells to fail, and there would be an abrupt drop in both the voltage and Ah.

Just as the nominal voltage doesn't tell the whole story about battery voltage behavior, ampere-hours is also squirrelly. The actual ampere-hours able to be extracted from a fully charged battery is a function of how many amperes are drawn and how they are drawn.

[5] There is a limit of 20 amperes.

To assess the SOH of a battery, it is essential to have the same test conditions. If I draw a constant current, I will get a different number of ampere-hours out of a battery than if I draw a varying current. I want to look at variations in battery capacity, not mix in variations in my test current effect. Therefore, my test circuit always presents the same load to the battery under test.

When the battery is new, I measure the number of ampere-hours I can extract as the voltage drops from full charge to empty. This is my 100% SOH ampere-hours. Periodically, I will run this test again and get a different number of ampere-hours. This will tells me the SOH. For example, if I initially draw 10.4 Ah and a few years later only get 5.2 Ah, the SOH has falled to 50%.

Appendix II: Software Details

I wrote the code in C/C++. You will notice that my function and variable names are descriptive. For example, I have the function

```
collectAndOutputData()
```

which, I do hope you see, collects and outputs the data.

Variable names include units and data format.

```
startTimeSecondsLong
```

This is the start time in seconds, and the variable is defined as a long. This verbose convention has headed off many software bugs.

Comments on almost every line of code further attempt to explain what is going on. However, if all I can say is already reflected in the name of the function or variable, I don't add any text.

During setup(), I configure my General Purpose I/O (GPIO) and the serial link back to the PC. Then I monitor the battery voltage and wait until it is greater than 20V. During this time, the LED toggles on for 1 second and off for 1 second. When the battery voltage is more than 20V, I take a series of readings while the battery voltage settles down. The highest value is my open-circuit voltage.

Next, I enter loop(). On each pass, I check the battery voltage. If it is below 40V, I disconnect the load to prevent damage to the battery[6].

[6] The loss of load current will cause the battery voltage to rise, and I do not want the load current to toggle on and off. Even with my defective battery, the voltage didn't rise more than 1V. Therefore, I only turn the load current back on when the battery voltage is above 42V. I can't think of a case where the battery voltage would dip down and come back up, but I don't want the program halting.

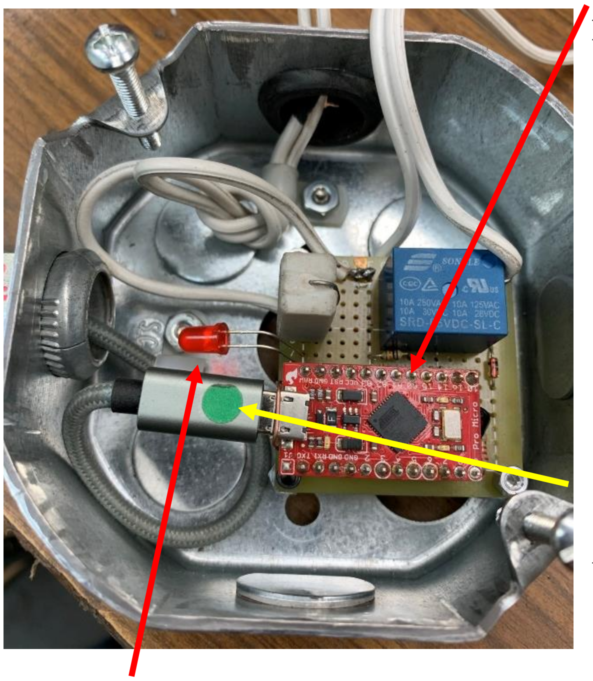
I toggle the LED on for 0.2 seconds and off for 0.2 seconds. This is useful because data only appears on the PC's screen every minute, and it is good to know the processor is sane.

Every minute, Vbat and the cumulative ampere-hours are read, processed, and sent to the PC. This ends when the battery voltage falls below 40V.

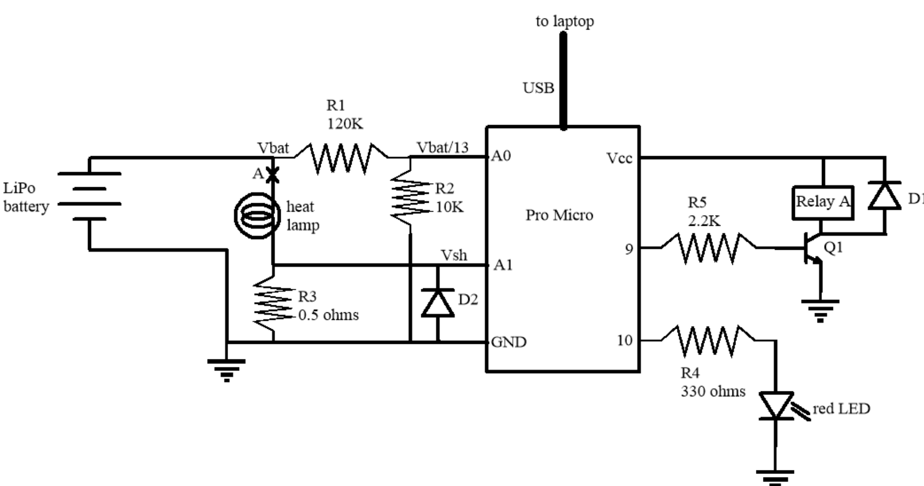
For more details, please see the code. The file is in ".ino" format so is ready to be loaded into the Arduino IDE.

See Sparkfun.com and search for Pro Micro if you need help loading this code.

Appendix III: Circuit Description



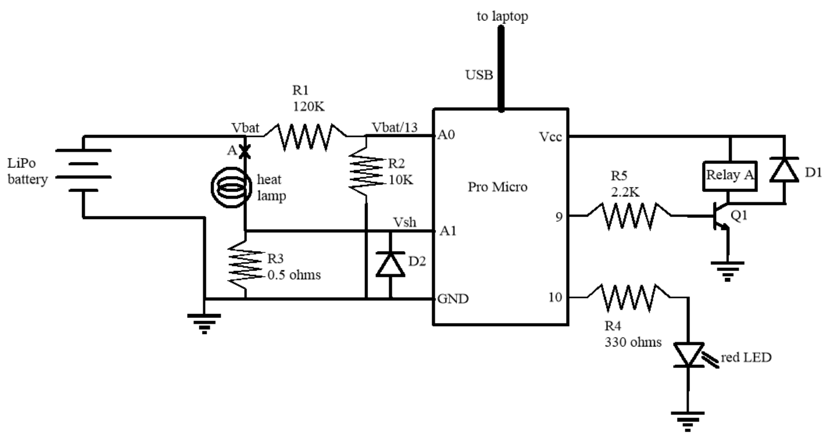
A Pro Micro processor runs the show. It measures V_{bat} and V_{sh} while keeping track of time and sends the data out a USB interface to a terminal emulator running on my laptop. Based on this data, the processor controls a status LED and a relay (blue box). The relay's contacts are in series with the heat lamp.



V_{bat}, the voltage from the LiPo battery, is divided down by R_1 and R_2. They form a 13 to 1 divider to keep the voltage at analog input A0[7] below 5V. V_{sh}, the voltage developed across shunt resistor R3, won't get above 1V, so there is no need to divide it down.

If the user connects the battery backward, R_1 in combination with a diode built into the Pro Micro, will prevent damage to input A0.

[7] All Pro Micro labels are logical, not physical. For example, the board is marked "10" but this is physically pin 13.



I added diode D_2 because a large current could flow out of input A1 if the battery was connected backward and relay contact A was closed. This can only happen if there is a software bug because V_{bat} would be less than 20V, so the relay should not operate.

While logical pin 10 is high, the red LED turns on.

The relay I chose draws more coil current than can be safely supplied by the Pro Micro. R_5 and Q_1 solve this problem. When logical pin 9 is high, only 2 milliamps flows through R_5. This is more than adequate to drive the NPN transistor into saturation. The relay then has V_{cc} minus about 0.1V across it. V_{cc} is 5V. When logical pin 9 goes low, Q_1 turns off. The current through the coil of the relay continues to flow so it needs a path. Without diode D_1, this path consists of Q_1 breaking down (not good).

I built the circuit on a 1.5” by 1.5” perf board. The Pro Micro is socketed. The shunt resistor is on end so air can circulate around it. It should dissipate about 1.1 watts.

The red LED is visible through the hole that passes the USB cable. In most cases, the user does not need to see this LED, so I didn’t bother to mount it on the outside of the enclosure.

I used a porcelain bulb socket out of an abundance of caution.

The bulb doesn't get very hot, but a plastic socket only saved me 20 cents.

Notice that I marked the terminals of the battery with minus and plus.

The strips of copper soldered to the SOHA wires are also marked.

Bill Of Material

Label	Value	Note
Pro Micro	5V/16 MHz	From Sparkfun.com
R1	120K 1/8W 5%	
R2	10K 1/8W 5%	
R3	0.5 ohm 5W 5%	
R4	330 ohm 1/8W 5%	
R5	2.2K 1/8W 5%	
Heat lamp	120VAC 250W bulb	Hardware store
Red LED	Any low power type	
Q1	Any low power NPN	
Relay A	Coil: 5VDC; normally open contact rated at 2 amperes 60VDC	
D1	Any low power diode	
D2	Diode able to pass 2 amperes	optional
porcelain bulb socket	Fits the heat lamp	Hardware store
Enclosure	4 inch steel gem box	Hardware store
USB cable		
Cable clamp for USB		
Cable clamp for battery wires		
Misc. fasteners		