hate to see folks suffer with old-fashioned remedies. After three decades of such anguish, I decided that enough is enough. So what am I talking about? Well, my focus for today’s pain relief is related to monitoring the battery packs used in RC models. The cure comes as BatMon, the sophisticated battery monitoring accessory shown in Photo 1.

I started flying RC model aircraft in the late 1960s. Back then, I was a young lad with an expensive hobby supported by neighborhood lawn mowing jobs. Because of my limited budget, I used inexpensive Rayovac carbon zinc cells in my RC transmitters and receivers. My radio gear was simple and this was a suitable solution.

But, affordable digital proportional radios eventually came along and the battery needs changed dramatically. Rechargeable NiCd battery packs became the power source of choice. The common configuration for airborne electronics was a four-cell pack that provided 4.8 VDC. Battery capacities were typically 500 to 600 mAh.

Checking the battery was something of a witch hunt. You would measure the pack with a voltmeter modified to expand the reading near the nominal 4.8-V range. A flashlight bulb was used as a moderate resistive load. A measurement that was under 4.7 V would indicate that it was time to end the day, because the pack was unsafe to fly. A higher voltage was assumed to be good to go. It was a reliable method if you observed its limitations, but at best it only offered a pass/fail status.

Other than by gut-feel experience, you never really knew the true discharge state of the pack. But given the needs of the day and the available technology, I was happy with the voltmeter test. Sure, new folks to the hobby would occasionally misuse the method (or ignore the test) and fly on a near empty pack. Even the pros did this from time to time. As you can guess, an airborne RC model with a dead battery is not a pretty sight. Fortunately, balsa wood was cheap back then.

Because a battery’s storage capacity is analogous to a car’s fuel tank, I wanted to be able to see the charge level in the same way a gas gauge works in a car. That would be much better than the voltage method, because a peek at the remaining capacity would offer advanced warning if I were flying on fumes, so to speak.

The underlying problem is that the discharge voltage curves for NiCd batteries are not linear. In fact, they spend nearly all of their useful discharge time at 1.2 V per cell. To complicate matters, the voltage is affected by the load (servo movement) and outside temperature. The voltage characteristics also change as the battery ages. Predicting the remaining battery capacity with any accuracy was out of the question for the average modeler.

Fast forward about 30 years. I now fly RC model helicopters. These high-tech aircraft are notorious for consuming battery current because the servos are quite active and under a heavy load. But, no matter what type of model is flown (or driven), I still use NiCd battery packs. Following them in popularity are the nickel metal hydride (NiMh) type. Both chemistries provide 1.2 V per cell and are rechargeable. Their nonlinear discharge curves have not changed much. There are other battery technologies that are in use, but these two are used by most RC hobbyists.

BatMon to the Rescue
A Battery Monitor for RC Applications

For years, hobbyists have relied on voltmeters and guesswork to monitor the storage capacity of battery packs for RC models. Now, Thomas introduces a more precise high-tech battery monitor that is small enough to be mounted in the cockpit of an RC model helicopter.
We still use four cells to power our receivers and servos. Some folks have implemented five-cell power sources to turbo charge their servo speeds. Over the years, the packs' milliamp-hours capacities have dramatically increased and the size and weights have tumbled. I am happy to report that cell reliability is extraordinary. Battery troubles nowadays are nearly always self-inflicted.

But, the same inaccurate voltage test is used to determine the discharge state of battery packs. Oh sure, the measuring devices have transformed into cute bar graph indicators, bright LEDs that blink morse code warnings, and audio beepers. Of course the expanded scale voltmeters that we used in the old days are still popular, too. But all of these methods rely on the same brainless go/no-go voltage test.

So, what gives? The battery gauging technology required to report the true remaining cell capacity [milliamp hours] has been around for years. This is common practice in portable computers and other consumer devices. Patiency, I waited for an RC battery gauge to show up at the hobby store.

Sadly, a little on-board gadget that would work with my RC receiver never materialized on the store shelf. Several years ago there was a rumor that one was available, but it quickly disappeared long before I could get my hands on one. Today, electric model hobbyists use the digital watt-meter devices, but they are designed to monitor the heavy currents consumed by electric motors. I wanted finer resolution so I could use it with my RC receiver and servos.

With that in mind, a couple of years ago, I convinced my firm that we should tackle this challenge. Although we were not in the RC equipment business, there was some interest. It was decided that I would develop a prototype that would act as a proof-of-concept. The premise was that if I could stir up some interest among local RC pilots, then a more advanced design would follow before we attempted to pitch it to the hobby industry. With luck, this was to become a low-cost RC accessory at the local hobby store.

As you will soon see, the project was completed and has served me well for over two flying seasons. Sadly, it did not become a commercial offering. I’m happy to report, however, that I finally have what I always wanted and I’m pleased to share it with all of my RC comrades. The project is well suited for monitoring the battery in nearly any electronic device, so its use is not limited to RC applications.

At the start of the project, I assumed that the most logical design centered on installing the tiny battery gauging IC inside the battery pack. A special hand-held LCD readout would plug into the model’s charge jack (a handy place to do so). It would extract the remaining capacity via the unused connector pin found on all RC receiver batteries. The nice thing about having the data stored inside the battery is that it allows you to swap the pack and the data remains with it. It also appeared to be a cost-effective arrangement.

But this solution was soon deemed unacceptable for several reasons. First, existing battery packs would need to be outfitted with the special IC. This is a low-cost effort at the time of pack assembly, but a retrofit connector-based solution would be more costly. Also, it was not common to swap packs in the field, so this assumed advantage really did not add much value. But most importantly, this concept did not have any visual warning features that would alert you of pack trouble while flying.

My solution evolved into the BatMon, a standalone device that can mount in each model aircraft (see Figure 1). This is not your typical larger-than-life Gotham City solution. It’s only 1.3” × 2.8” and weighs one ounce. But the BatMon does have the typical dual persona expected of a super hero.

For user simplicity, it reports battery capacity as a zero to nine (0% to 90%) level value. This is my favorite mode because it works just like a car’s gas gauge. However, for those of you who must see hard numbers, it also reports the actual remaining capacity—up to 2500 mAH—with 5% accuracy.

In addition, it reports problems associated with battery pack failures, bad on/off switches, and defective servos. A super-bright LED indicator flashes if any trouble is detected. Even in moderate sunlight this visual indicator can be seen from a couple hundred feet away, which is perfect for fly-by checks.

The BatMon is compatible with all of the popular battery sizes. Pack capacities from 100 mAH to 2500 mAH can be used. They can be either four-cell or five-cell of either NiCD or NiMH chemistries. The battery parameters are programmed by using a push button and simple menu interface.

The device has three connections. Two of the RC-style connectors are inserted in series with the battery pack. Because you are making current measurements, this sort of intimate connection is necessary. There is a third connector that plugs into a spare channel of the RC receiver (or you can use a Y adapter on any servo). This connector enables the display when the RC receiver’s power switch is turned on.

The thought of losing control of a flying model, due to a defect in the BatMon, was not something that I wanted to encourage. My main concern was that the BatMon circuitry was directly in the path of the RC equipment’s battery. Fortunately, only three passive components are in this critical area, and two of them are RC type connectors. The third is a shunt resistor that was carefully selected to ensure reliable operation. If you assemble the BatMon correctly, it will not pose a risk to your equipment’s reliability.

Perhaps the most important element of the project was the display. I searched for a suitable two-line LCD,
but I didn’t find any decent offerings. I decided to use a seven-segment LED display for the initial design. I reckoned that a custom LCD would need to wait for a production version of the BatMon. In retrospect, the LED display works surprisingly well.

The seven-segment display will accommodate one character at a time. This works nicely with the zero to nine battery level values, but some trickery is used to show the four-digit capacity and two character error codes. These values are merely “spelled out” in a flash card sort of way. For example, 575 mAh would repeatedly flash as “5,” “7,” or “5.”

An issue with the LED readout is that it consumes almost 60 mA of current. Because it does not need to be on while the aircraft is flying, it can be set to display only when you press the push-button switch. Upon release it will shut off after a few seconds. As a convenience, the level value flashes every 5 s when the display is off. But, if you want the display active at all times, you can set the BatMon to do so.

The battery gauging IC that I used is from Dallas Semiconductor. There are other firms that have similar parts (Unitrode, TI, etc.), but the Dallas DS2438 Smart Battery Monitor was a perfect choice for my RC application (see Figure 2). This eight-pin coulomb-counting chip contains an A/D-based current accumulator, A/D voltage convertor, and a slew of other features that are needed to get the job done. The famous Dallas one-wire I/O method provides an efficient interface to a PIC16C63 microcontroller. There’s a lot I could say about the DS2438, but the Dallas folks have done a fine job of explaining its operation in the 30-page datasheet. But, I will gladly take you for a quick stroll just to help acquaint you with some areas of interest. You will quickly realize that the chip has significant smarts built into it.

Looking at the block diagram in Figure 3, you can see that there are few connections to the outside world. Communication to the host requires only one pin. Besides a host microcontroller, there are only three external components required to use the DS2438. RSENSE is a fractional ohm current sense resistor. RF and CF are configured as an input filter. Don’t let this simplicity fool you, this fellow has a lot going on under the hood.

The DS2438 is a register-based device. The registers are accessed by the host through a clever bidirectional communication scheme. If you have ever used a low-pin-count Dallas IC before, then you are no doubt acquainted with the company’s one-wire bus protocol. The chip is self-sufficient and can operate by itself after the host has configured it. A nifty one-wire trick is that the registers can be accessed even if the battery source is completely dead. This feature is not needed in the BatMon design.

In the BatMon, the one-wire bus begins at pin 6 (port RA4) of the PIC16C63 microcontroller and terminates at the DS2438’s DQ I/O line (pin 8). Using bit-banging I/O, the PIC can read and write the necessary registers. The timing is critical, but the PIC is capable of handling the chore. Full
MBasic Software
Whether your just learning or a professional, Programming PICmicro® MCUs has never been easier! MBasic is much simpler than C or Assembly. MBasic creates a one click solution that allows you to experiment and test code changes on-the-fly! Bring your projects to life quicker and easier with MBasic for PICmicro® MCUs!

ICD - In Circuit Debugger
Stop wasting time strategically planting debug statements throughout your entire program. MBasic includes a built-in ICD (In Circuit Debugger) FREE. Watch variables, SFRs and RAM values as each line executes. MBasic’s ICD is so easy to use, even the first time user can have it up and running in minutes!

MBasic and Assembly Language
Seasoned Assembly programmer or just learning? MBasic allows an easy mix of BASIC and Assembly. Instead of spending weeks writing your program, have it done in days!

Syntax
A complete set of easy to use commands! Serin, Serout, If..Then..Elseif..Else..Endif, Do..While, While..Wend, OWin, OWout, ADin, Pulsin, Pulsout, PWM, Xin, Xout and more! Plus easy access to all PICmicro built-in hardware peripherals.

Math
MBasic is the only BASIC compiler that supports 32 bit floating point and integer math. This includes 32 x 32 bit divides and multiplies.

Download MBasic Lite FREE!
Try before you buy!

MBasic Pro Only $229.95
Includes Printed Manual and CD-ROM with MBasic.

ISP-PRO Programmer
- Uses PC Serial port (USB w/ adapter)
- Very Simple to use
- Free Software updates
- Complete with Windows IDE!
- Easy in-circuit programming
- Supports PIC, Scenix, I2C and more!
- Firmware upgrades Free!
- RJ-11 / 10 Pin Header
- Includes RJ-11 Cable

Optional ISP-PRO Items Available
- Plastic Enclosure
- 40 Pin ZIF Adapter
- Universal Adapter

Solderless Development
- Completely Assembled Board
- In Circuit Programmable (ISP)
- Solderless Bread Board
- Built in RS232 (w/ Max232)
- Built in Power Connector
- Removable Oscillator
- Documentation
- Auto Disconnect for RB3, RB6, & RB7
- Available in several models

Development Kit
Includes:
- MBasic Pro Compiler
- 2840 Development Board
- ISP-PRO Programmer
- PIC16F876-20
- 10Mhz Resonator
- Serial Cable
- Power Supply
- CD-ROM and Manual

Only $59.95 Starting At $59.95 Starting At $159.95

Join our on-line PIC forums, information and help FREE!!!

To order visit www.basicmicro.com or call 734-425-1744
M-F 10 AM to 6 PM EST.

MBASIC is a registered trademark of Basic Micro Inc.
PICmicro is a registered trademark of Microchip Inc.
details of the I/O protocol are outlined in the DS2438 datasheet, so I won’t repeat them here.

The integrated current accumulator (ICA) register is the main workhorse inside this petite part. It records current consumption using a coulomb-counting scheme. An internal oscillator samples the voltage across the RSENSE resistor (R12) at a rate of 36.4 times per second. The RC filter (R2 and C2) has a cutoff of about 16 Hz, which tames noise but allows the needed current spikes to be captured.

Because small current ADC offset errors can severely affect the long-term accuracy of the ICA, the DS2438 has a register that cancels the offset currents. The offset register makes the correction. The BatMon is designed to use this clever nonvolatile R/W register. There is a special user-invoked mode that initializes the correction value. This is done using the push-button switch after the board is assembled. For full details to the offset calibration, read the user guide, which can be downloaded from the Circuit Cellar ftp site.

The DS2438 has registers that report instantaneous current, battery voltage, and external voltage. These features were a blessing to the project. But not all of the goodies were needed. The temperature, elapsed time meter (ETM), disconnect time stamp (DT), charge current accumulator (CCA), discharge current accumulator (DCA), and end-of-charge (EOC) registers are all ignored. These features are available if you wish to expand the functionality of the BatMon. For example, cold battery temperatures will reduce effective capacity, so the temperature register could be used to adjust the reported values.

After the PIC has initialized the DS2438, battery consumption is automatically tallied by the ICA register. It has eight scaled bits of resolution, so the sense resistor (R12) determines the bit weights. With the selected 0.050-Ω value, each count is 9.765 mAH, which results in a max count of 2500 mAH.

The ICA can sense the direction of current flow, so the accumulator counts down during battery discharge and counts up during recharge. The ICA register is read by the PIC16C63 every 100 ms and the value is scaled into milliamp-hours and a zero to nine level. You decide what to display on the seven-segment LED. You can toggle the two measurements (level versus capacity) by pressing the push-button switch.

I also wanted to accommodate some of the oddities that are associated with NiCd and NiMh batteries. For one thing, they are not 100% efficient during the charge cycle. They also tend to self-discharge when they are resting. To partially mask these issues, the PIC compensates the values held in the ICA register. During a charge cycle, the PIC reduces the ICA’s value by about 20%. This mimics an 80% charge efficiency. During idle periods, the ICA is reduced by 2% every 24 h.

Both of these corrections are practical values and work well. Zealots may argue that the values are too generalized, but hey, these are often the same guys who use a voltmeter to check their packs. Besides, the BatMon does not change the way you maintain your batteries, so a fresh charge is always expected before using them. This requirement makes the correction process unnecessary, but I added it to the software anyway.

Limiting the BatMon’s job to mere fuel gauging would have been shortsighted. After all, there are all sorts of things that can go wrong but can easily be detected by monitoring the pack’s voltage under load. The DS2438 can measure two voltages, so it was elected to lend a helping hand.
MSP430 offers 50X more throughput.

MSP430—the choice in ultra-low-power Flash MCUs

Experience the ultimate SOC solution for battery-powered measurement. A flexible clock system switches from ultra-low-power standby to high-performance signal processing in less than 6 μs. Embedded emulation reduces design cycle time. Get your design started today with the easy-to-use MSP-FET430P120 Flash emulation tool.

<table>
<thead>
<tr>
<th>Device</th>
<th>Flash</th>
<th>RAM</th>
<th>I/O</th>
<th>WDT</th>
<th>Timer_A</th>
<th>USART</th>
<th>ADC</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSP430F1232</td>
<td>8 kB</td>
<td>256</td>
<td>22</td>
<td>✓</td>
<td>✓</td>
<td>1</td>
<td>10-bit</td>
<td>$2.79</td>
</tr>
<tr>
<td>MSP430F1222</td>
<td>4 kB</td>
<td>256</td>
<td>22</td>
<td>✓</td>
<td>✓</td>
<td>1</td>
<td>10-bit</td>
<td>$2.62</td>
</tr>
<tr>
<td>MSP430F1132</td>
<td>8 kB</td>
<td>256</td>
<td>14</td>
<td>✓</td>
<td>✓</td>
<td>–</td>
<td>10-bit</td>
<td>$2.48</td>
</tr>
<tr>
<td>MSP430F1122</td>
<td>4 kB</td>
<td>256</td>
<td>14</td>
<td>✓</td>
<td>✓</td>
<td>–</td>
<td>10-bit</td>
<td>$2.24</td>
</tr>
</tbody>
</table>

MSP-FET430P120 development tool—$99

Features
- 200-kSPS 10-bit ADC with Data Transfer Controller
- 8-kB ISP Flash
- 0.8-μA standby mode, 250-μA active mode
- On-board brown-out detector
- USART can be used in UART/SPI™ mode

www.ti.com/sc/hipa7554u
1-800-477-8924 ext. 7554
The PIC reads the two independent voltage measurements and determines if they are within safe limits. The battery voltage is measured at the DS2438’s V+ and V- pins. The battery pack is a low-impedance current source, so the voltage drop is small under the expected loads. However, if the voltage is too low, the display will report a U1 error [battery voltage low]. This can trap charge issues, cell failures, or serious current draw problems from the servos.

The other measured voltage is on the switched side of the RC system’s on/off switch, complements of the DS2438’s VAD input [pin 4]. Because this measurement is made on the RC receiver’s load side, it will see a higher nominal voltage drop. If it is excessive, a U2 [bad switch, weak battery, stalled servo] or U3 [low battery voltage, binding servo] error will be reported.

If any trouble is detected, the LED status indicator will flash. This LED can be installed on the PCB or remotely mounted on the aircraft. It is bright, so you should be able to see it even if the aircraft is flying. It will flash a single wink if the battery capacity is low [under 30%]. A double-wink indicates a more serious issue that needs immediate attention.

One of the issues that needed to be addressed was that the end user’s battery pack ratings can vary. To work effectively, the BatMon needs to know the pack’s rated voltage and milliamp-hours capacity. After all, the model could be equipped with nearly any pack capacity. To complicate matters, four cells [4.8 VDC] or five cells [6 VDC] might be installed. The solution involves the push-button switch and some fancy footwork.

There is a special programming menu that can be activated by following a certain power-on sequence. First, turn on the model’s power using its on/off switch. Wait for the display to appear. Second, turn off the power. Immediately press and hold the push-button switch [SW1]. Third, within 2 s, turn on the power and confirm that the display shows a flashing “P.”

From this point you can traverse the different menu settings by quickly cycling the on/off switch. Items in a menu are selected by pressing the push button. You can choose cell capacity [100 to 2500 mAh], cell count [four or five], and other features. Full details are contained in the user guide.

The PIC16C63 firmware was written in C using CCS’s PIC C Compiler. It uses all but about a dozen bytes of...
the 40-KB code space. If you wish to add features, you should plan on upgrading to a different MCU.

All of the important tasks are performed by the real-time kernel, which is done by the serv_jiffy() function (a simple task scheduler that’s synchronized to Timer 1). It has 50-ms resolution and can launch sequential tasks at periods up to once per minute. It is called from the main function in order for the PIC’s sleep cycle to easily utilize it.

Speaking of the sleep cycle, when the equipment is turned off, the PIC is shut down after a few seconds of being idle. The operating currents are reduced to ~180 µA while in Sleep mode. To achieve this low current, you must disable the brownout fuse during chip burning.

During Sleep mode, the PIC wakes up periodically to calculate the self-discharge and charge correction values. It is also during this period that it checks to see if the RC power switch has been turned on. If so, the PIC fully wakes up, enables the display, and performs all of the monitoring features.

The software is fully documented. It will not be easy to dissect my code, however, all of the information to do so is in the source text. Given the size of the program and some of its complexities, I’m sorry to say that I will not be able to answer specific questions concerning the software’s C functions. Besides, there’s no fun in being spoon-fed all of the answers. So, just roll up your sleeves and study the code.

The BatMon is not a good candidate for perfboard construction. A big issue is that RC models present a harsh operating environment. Vibration and less than pleasant landings demand that you use rugged electronic assembly techniques. My vote is that you design a circuit board for it. It is not a complicated circuit, so with the help of a freeware PCB program you should be on your way. My latest version used nearly all through-hole parts. I could have accomplished drastic size reductions if I used the SMT components.

Please be aware that sloppy PCB layouts may allow the 3.58-MHz oscillator to radiate unwanted harmonics. This could cause interference with RC receivers. Lastly, the current sense resistors’ accuracy can be compromised if you are not careful how it is connected to the DS2438. You can expect inaccurate capacity values if the sense line traces are in the current carrying path.

The connections to the battery pack and receiver are made with standard RC hobby servo connectors. They are available at most RC hobby shops. You will need a 22-AWG, two-conductor female cable for the battery (J1), a 22-AWG, two-conductor male for the RC switch (J2), and a three-conductor [any AWG] for the Aux In (J3) connector. Please note that the battery cables are heavy-gauge to minimize voltage drops. Keep them short (less than 6”) and strain relief all cables at the PCB.

I found that large heat shrink tubing makes a robust enclosure for the finished unit. The tubing I used is the type that is designed for RC battery packs. It is low cost and is sold by the foot at most hobby shops that specialize in RC cars. You can also use the heat shrink tubing that is sold as covering material for the main blades on RC model helicopters.

All you have to do is cut small holes for the three cables, slide it on, shrink it, and then trim some openings for the LED’s and switch. With some additional trimming, I can create flaps on the open ends that are easily bent down and glued in place. The finished unit is protected from field dust and it’s also moderately fuel proof (glow fuel is very conductive).

The finished unit is mounted in the model’s cockpit using double-sided tape or held with rubber bands (see Photo 2).

As you can see, there is a high-tech solution to monitoring your RC battery packs. If you’re still sold on the old-fashioned voltage method, be my guest to continue the practice. But, holy cow! With the BatMon, there is a better way. ☀️

Thomas Black designs and supports high-tech devices for the consumer and industrial markets. He is currently involved in telecom test products. During his free time, he can be found flying his RC models. Sometimes he attempts to improve his models by creating odd electronic designs, most of which are greeted by puzzled amusement from his flying pals.

SOFTWARE
To download the software and users guide, go to ftp.circuitcellar.com/pub/Circuit_Cellar/2002/143/.

SOURCES

PIC C Compiler
Custom Computer Services, Inc. (262) 797-0455
www.ccsinfo.com

DS2438 Battery Monitor
Dallas Semiconductor, Inc. (972) 371-4000
www.dalsemi.com

PIC16C63 Microcontroller
Microchip Technology, Inc. (480) 786-7200
www.microchip.com

Wishing you all the best in your flying endeavors!

Thomas Black