Using a capacitor to sustain battery life

April 28, 2014

31

My wife thinks I'm crazy.

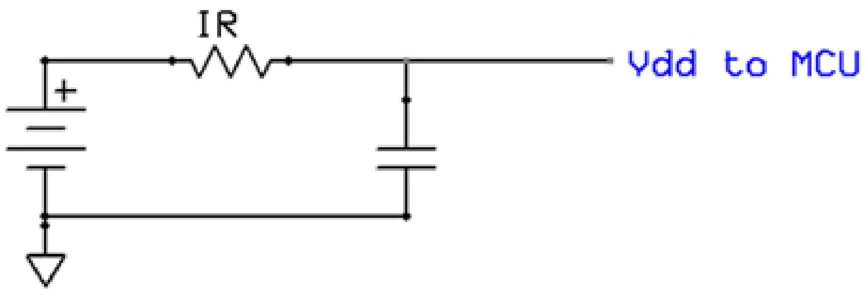
She has been watching me discharging piles of coin cells over the last year, collecting millions of data points about their behavior. “Honey, exactly why are you spending all that money on batteries you're just going to discharge?” It is pretty nutty, but the results are interesting, and dispel a lot of beliefs about what we can expect in long-lived, ultra-low-power, microcontroller-based products.

Last week I wrote about the behavior of CR2032 coin cells in applications that have to run for a decade or so. My experiments quantified the batteries' increasing internal resistance as they're used. The net result is that, depending on the system's current needs, the cell will appear dead long before it really is. That is, there will be plenty of capacity left, but none of it will be useable by the system.

Several thoughtful readers wondered if adding a capacitor across the cell's terminals could provide a short-term boost that could sustain a pulse load. It's not hard to show mathematically that the answer is “yes.”

But the math is irrelevant. In engineering we're always faced by conflicting needs and what appears to be a simple solution sometimes isn't.

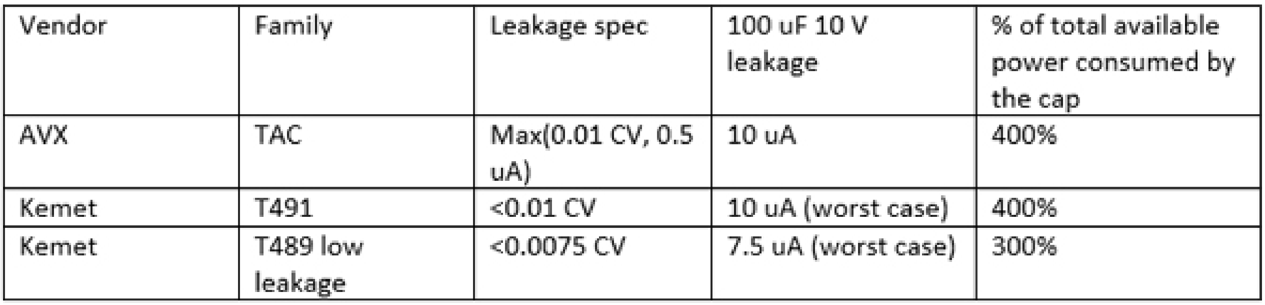
It's useful to think of the battery as a zero impedance source with an annoying internal resistor between it and the terminal. That resistor's value increases as the battery capacity is drained. If it were 50 ohms and the system needed 10 mA, the terminal voltage is down by half a volt, often enough to cause the MCU to crash. Add a big capacitor, as in the following diagram, and a short pulse load can draw on the charge stored in the cap.



One reader cited a TI white paper (Coin Cells and Peak Current Draw) where the authors conclude that this is a viable solution. They show a case where a capacitor must sustain a 30 mA load for 1 ms. An 87 uF capacitor will do the trick, though they recommend 100 uF since no one makes 87 uF devices.

There are a couple of problems with this conclusion.

First, the capacitor leaks. It's always across the battery terminals, so is sucking juice, depleting the battery, even during long sleep intervals. How much leakage? It depends on a number of factors, including dielectric material. Let's consider tantalums, which offer great capacitance versus cost figures. The following table shows the numbers for a 100 uF part. (For tantalums it's usual to rate leakage in CV, where V is the rated, not the applied, value.)

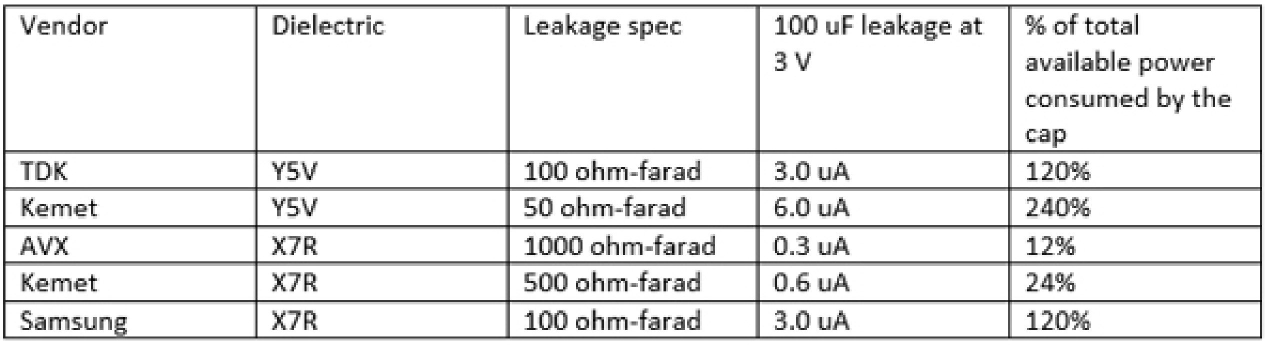


Vendor	Family	Leakage spec	100 uF 10 V leakage	% of total available power consumed by the cap
AVX	TAC	Max(0.01 CV, 0.5 uA)	10 uA	400%
Kemet	T491	<0.01 CV	10 uA (worst case)	400%
Kemet	T489 low leakage	<0.0075 CV	7.5 uA (worst case)	300%

The last column is the most telling. I started this series showing that MCU vendors who claim decades of potential operation off a CR2032 are completely off-base, and demonstrated that the longest life one could hope for is a decade (though few will ever achieve that). A CR2032 offers about 220 mAh of capacity, which means the average current draw over 10 years cannot exceed 2.5 uA. Just the capacitor's leakage will suck the battery dry in a fraction of a decade.

Who would have thought that a cap could leak more than Edward Snowden?

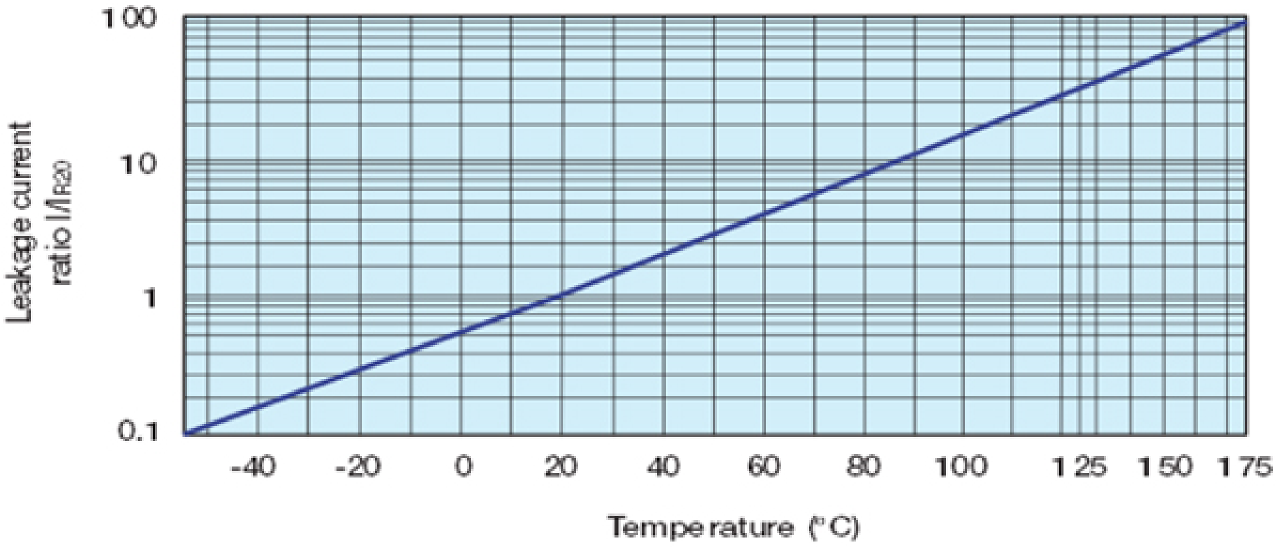
How about a better part? The best low-leakage capacitors with the C values needed are MLCC. MLCC leaks are specified in ohm-farads:



Vendor	Dielectric	Leakage spec	100 uF leakage at 3 V	% of total available power consumed by the cap
TDK	Y5V	100 ohm-farad	3.0 uA	120%
Kemet	Y5V	50 ohm-farad	6.0 uA	240%
AVX	X7R	1000 ohm-farad	0.3 uA	12%
Kemet	X7R	500 ohm-farad	0.6 uA	24%
Samsung	X7R	100 ohm-farad	3.0 uA	120%

It’s clear Y5V dielectrics can’t be used. Select a more expensive X7R device. And, one must be careful which X7R is selected as some leak as badly as Y5Vs, though exhibit better temperature stability, which has always been the primary reason to use X7Rs.

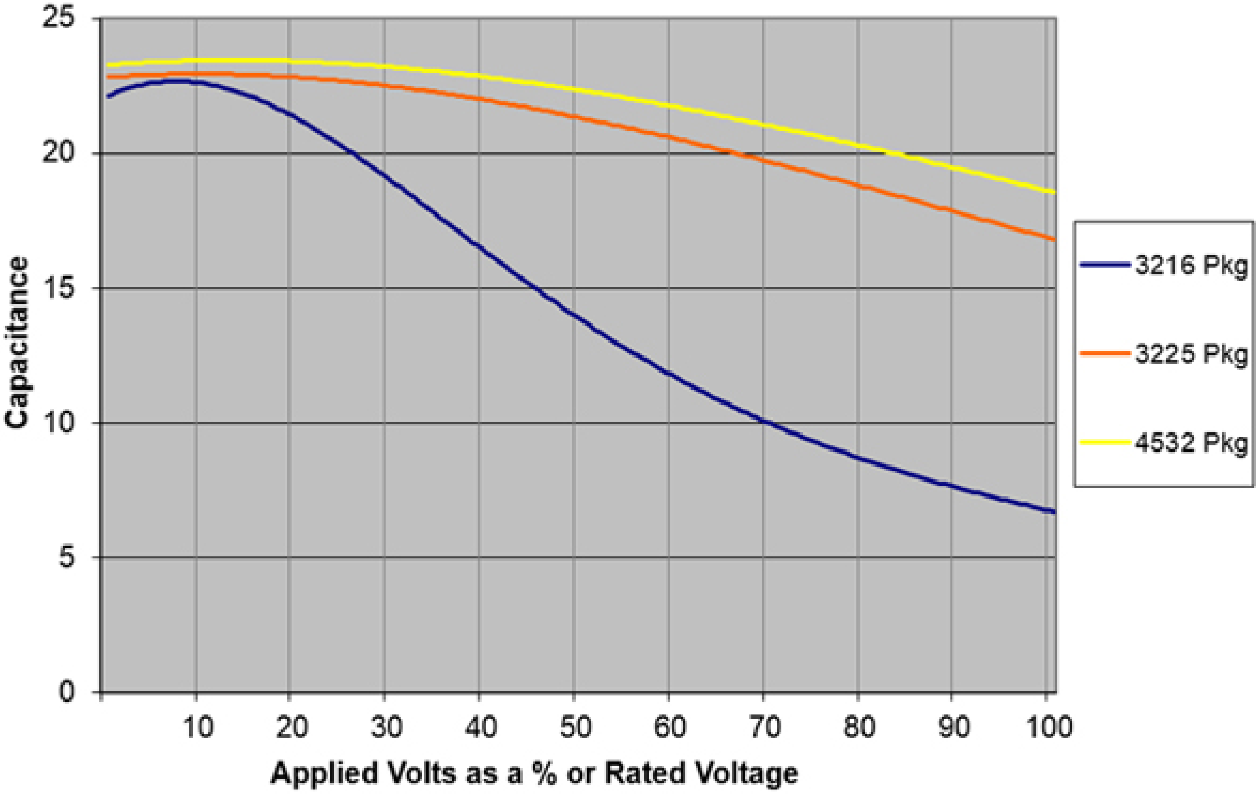
X7Rs are indeed very temperature stable, but not as a function of leakage! Here’s a typical graph of normalized leakage versus temperature:



A system that has to run over wide temperature extremes may leak two orders of magnitude more than shown in the previous table; that sweet AVX spec’d at 0.3 uA suddenly looks like a moderately low-value resistor. But even over room temperature, say 20 to 30 degrees, plan on twice the parasitic drain than specified. In the case of the AVX a quarter of the battery is lost to the capacitor.

It gets worse.

MLCC devices derate themselves. As the applied voltage increases, the effective capacitance decreases. Murata has a tool that helps model this, and here’s the result for 22 uF parts:



If space is precious you're screwed. Apply 3 volts to a nice small 6 V 3216 device and only half the capacitance is available. The TI paper specifies a 100 uF part, but in this package you'd have to use 200 uF. Alternatively, one could use a higher-voltage (bigger and more expensive) device, or use a larger package as shown in the graph.

Capacitors have tolerances, and though X7Rs typically are a good +/- 20%, one still has to figure that in:

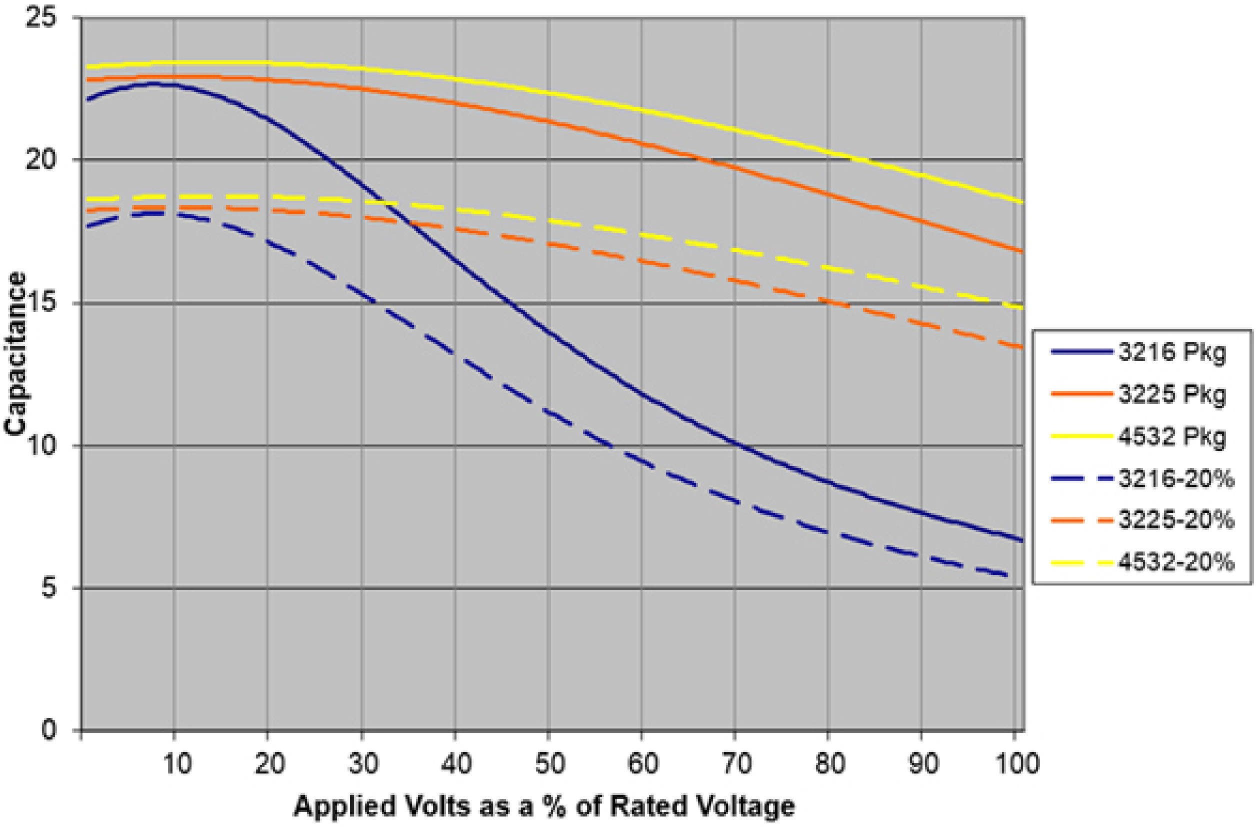


Figure on an even bigger capacitor to deal with tolerance.

The following graph shows the required capacitor, at 20 degrees C, for 10, 20 and 30 ms pulses with various loads, ignoring all of the complicated effects noted above. You'll have to factor all of those parameters in as well, which, as we've seen, can more than double the required uF. The leakage numbers are based on that AVX component. Though the TI paper uses a capacitor to boost power for 1 ms, for Bluetooth and other protocols tens of ms are more likely.

So we've done the math, and have figured out what size capacitor to buy. Let's ignore all of the unpleasantness and assume that the 100 uF part fits the bill, and that we're using a low-leakage AVX part. They're $14.50 each! Neither Digi-Key nor Mouser has them in stock, and quote a 25 week lead time. Digi-Key does have a 50 V version available today, but that will set you back $36.54.

A complete Raspberry Pi with 700 MHz ARM processor and half a gig of RAM costs less than that capacitor.

Maybe this is a government project where costs don't matter. Using the graph above we pick off a 400 uF part for a 10 mA 20 ms load (remember, this is before all of the derating that must be done). The leakage will eat about half the battery capacity. And, the capacitor doesn't exist; the biggest such part on the market is 220 uF. You can't buy a bunch of smaller parts and parallel them, since the leakage will go up.

What about a supercapacitor? These are astonishing devices that offer farad levels of capacity. Alas, the lowest-leakage supercap I can find is from Cap-xx, and they quote 2 uA at 23C, doubling at 70C. That's impractical since even at room temperature it'll eat just about the entire 2.5 uA current budget.

To add another wrinkle, don't forget that every MCU vendor requires one or more decoupling capacitors, even when running from a battery. Generally they want one of about 10 uF. Be sure to use a low-leakage part, and factor that into your battery capacity calculations.

Summing up, in the use case I've described (ten years of system life from a coin cell) it's generally impractical to get a pulse boost of Vdd from a capacitor across the battery. However, there are other ways to stretch battery life, which I'll cover in coming weeks.