Joule thief efficiency study, rev 1/ Fernando Garcia May 2009

There is plenty of information on the web with respect to the joule thief, that wonderfully simple and almost foolproof circuit. But this simplicity hides many subtleties. And like the saying goes, the devil is in the details. Although there is little that can go wrong, subtle changes can improve or decrease its efficiency. But before I start, the following disclaimers:

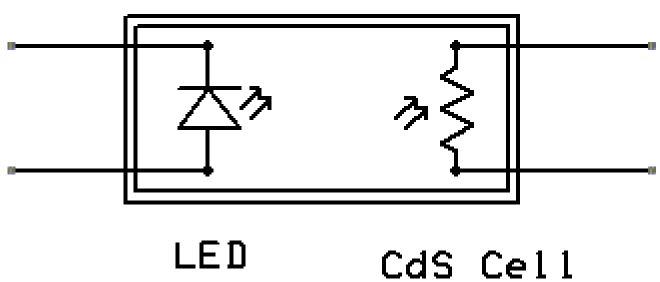
I performed this study out of technical curiosity exclusively; I don’t stand to gain or profit anything; nor do I have an axe to grind against or a bias in favor of any particular circuit. I’m publishing this as a way of saying ” thanks” to this forum which has given me so much, and hopefully this will spur some healthy discussion.

The tests shown here were performed with hobbyist-grade equipment, which has its limitations. Of course, laboratory-grade equipment will yield more accurate results. Having said that, I’m quite familiar with laboratory practice, and I’m pretty confident that the results presented are consistent, within the equipment limitations.

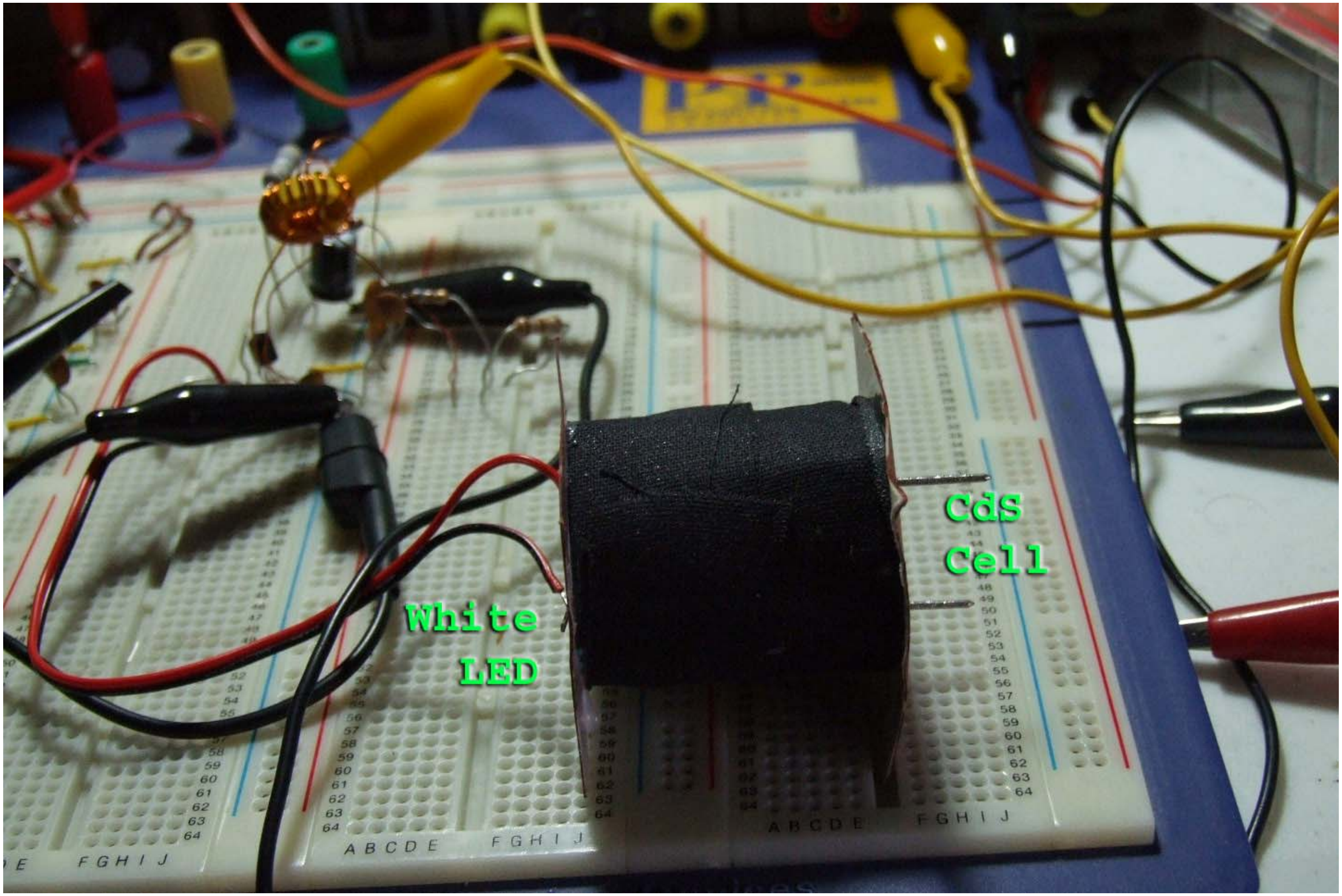
When one measures efficiency, one is measuring the ratio of an output variable against an input variable. The optimal method would be power output (volts times current) vs. power input (volts times current). In the latter case, since I’ll be holding the input voltage constant with a regulated power supply throughout the experiment, then I can simplify and measure only DC input current; In the former case, since the circuit drives an LED, it is best to measure light output.

This presents an interesting problem, since I don’t have the luxury of owning a luxmeter. However, I had previously purchased a surplus high quality CdS photocell. CdS photocells have the advantage that its luminous response approximates that of the human eye. By placing the LED in front of the photocell, and by reading the latter’s resistance, a relative light output measurement could be made.

To prevent ambient light from interfering with the measurements, I installed both the white LED and the photocell in a light-tight cylinder, made of white construction paper, which would ensure that all the LED’s light would be reflected into the cell. And to make it as hermetic to light as possible, I wrapped the exterior in black electrical tape.
This is the light box schematic.

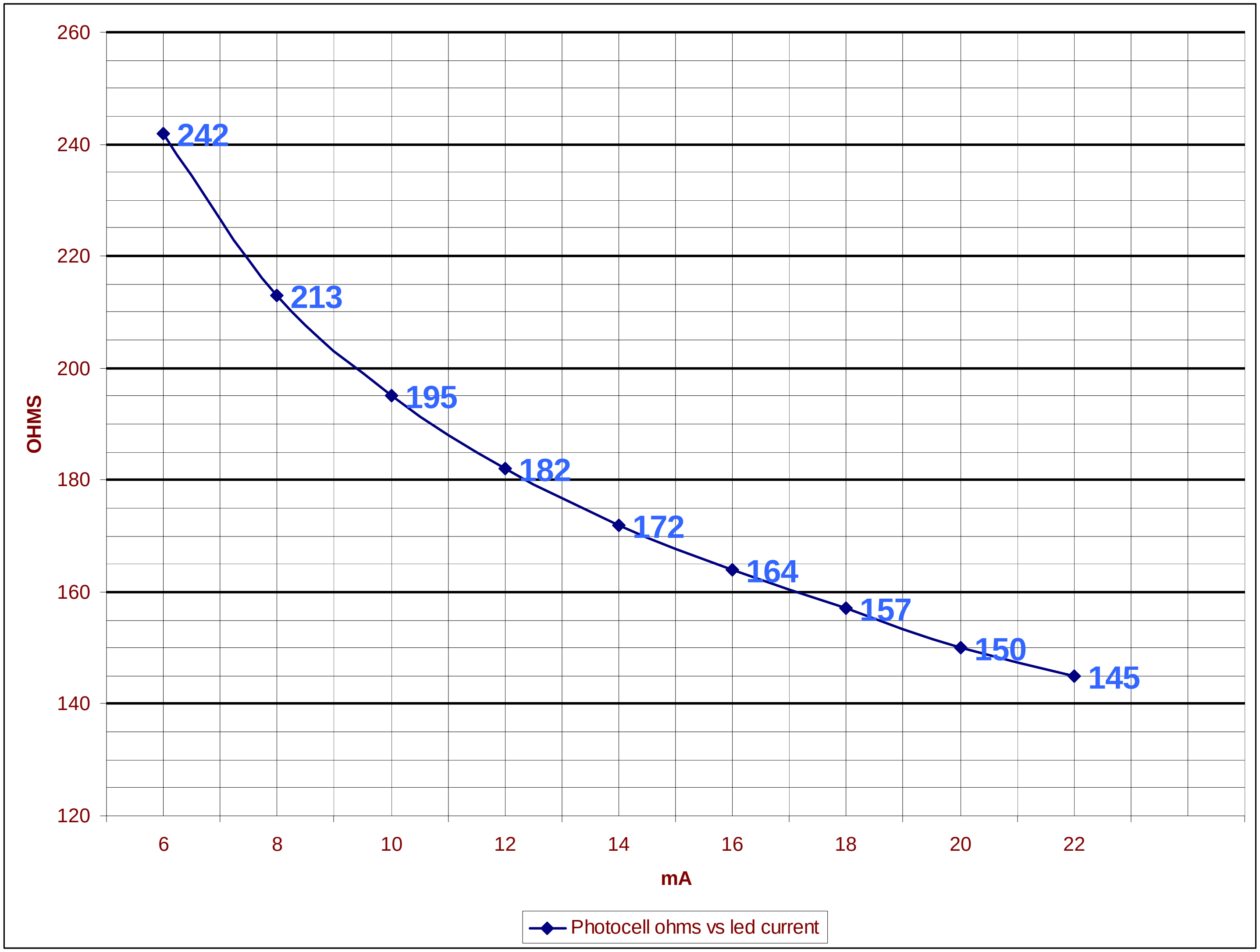


And this is a picture of the actual light box.



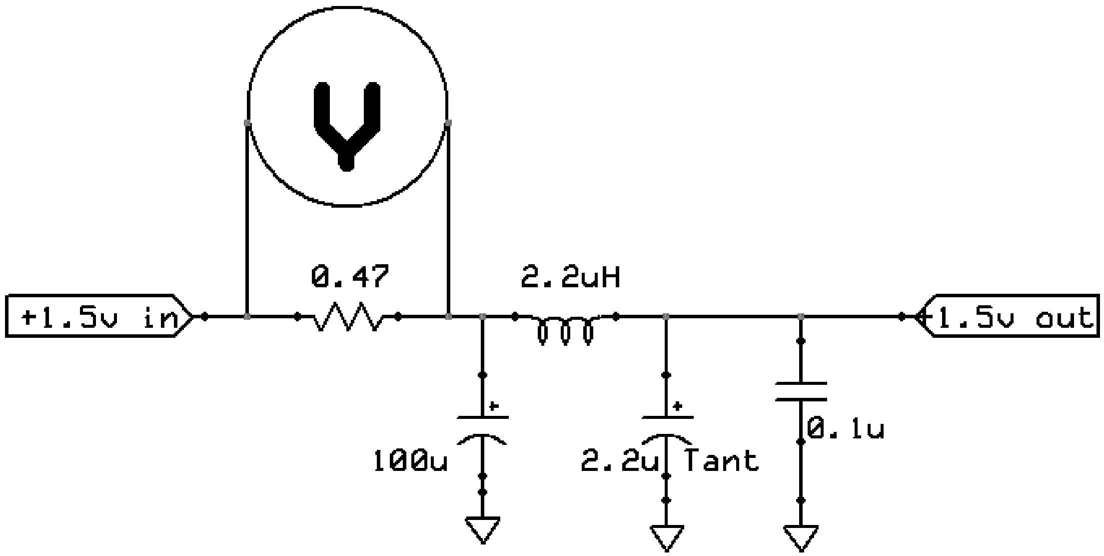
Once that the light box is built, we need to characterize its transfer function, in term of Cell resistance vs. LED current. We do this with pure DC from a variable power supply, where we monitor the LED current in 2 mA increments, and read the corresponding photocell's ohms. We can now plot it as shown below. Then, when we test the different joule thief's circuit variations, we again read the cell's resistance and find an <u>equivalent LED DC current</u>. For instance, if we read 185 ohms, the equivalent LED DC current would be approximately 11.5 mA.

We need to do this because the output current of the joule thief may have substantial ripple, and that ripple may have an effect on the actual light output, which is our desired output.

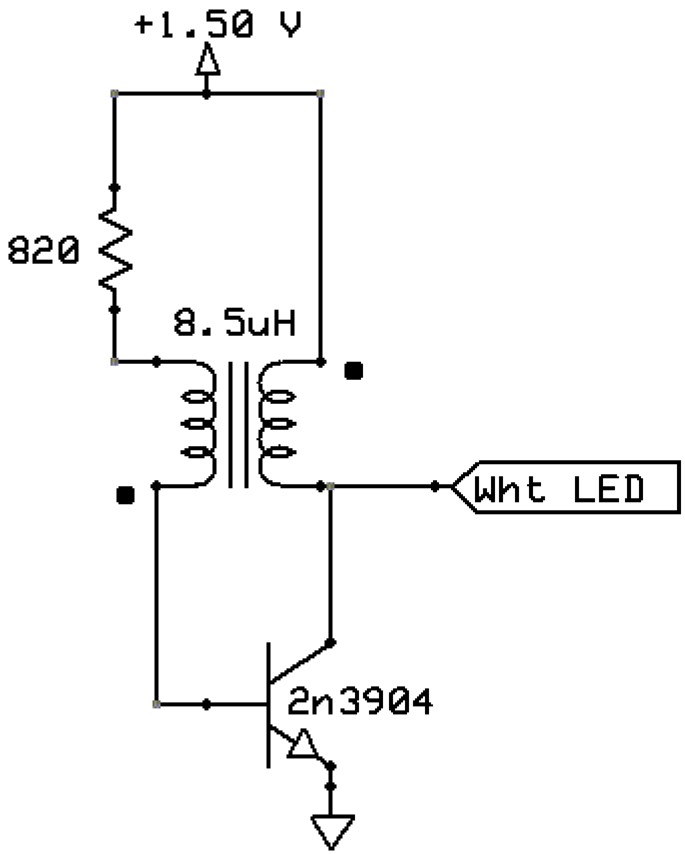


Next, we need to read the input current. Most web references suggest a low value resistor, and measure the voltage drop across it. Additionally, we need to average out the current, since it contains large amounts of high frequency ripple, that would throw meaningless results in a normal DMM.

The following circuit accomplishes both things



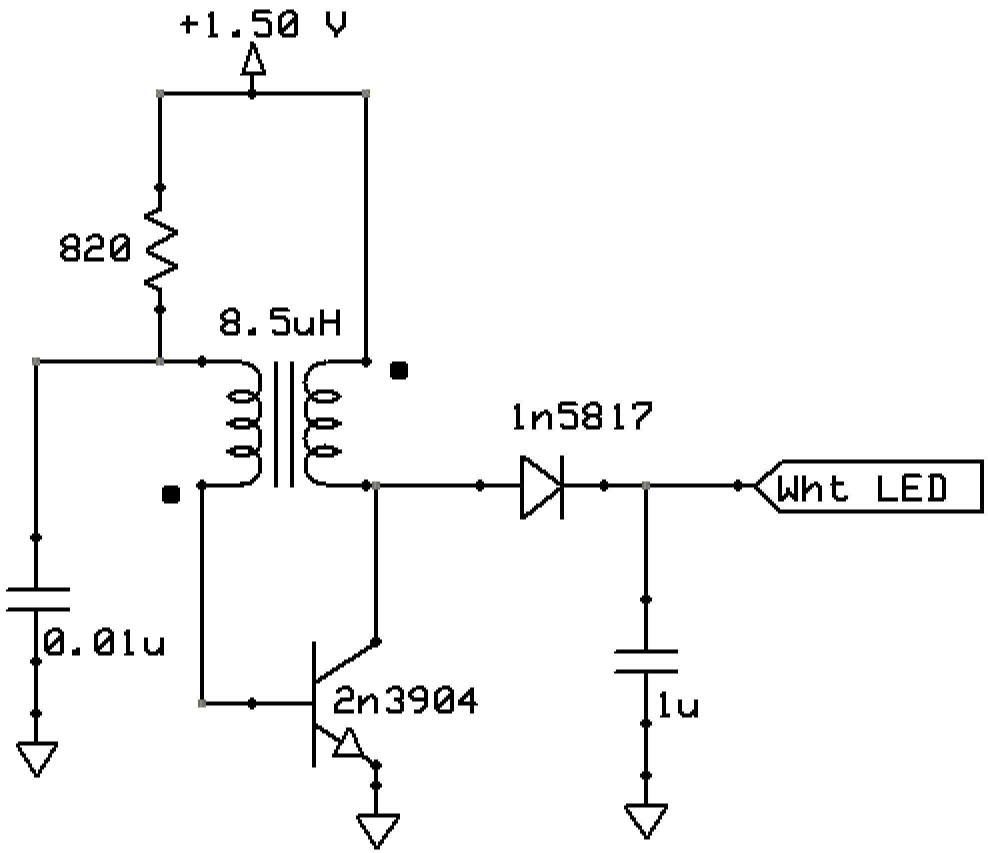
Voltage into the joule thief was always kept at a constant 1.50 volts.



This is the original Joule Thief. I employed a 2N3904 transistor, but clearly it is not optimal. I know for instance that the BC337 provides a higher light output….however I decided to stick with the 2N3904 because it is as plain vanilla as it gets and because it can be had for pennies.
Most web circuits indicate a 1 kohm base resistor, however, I found that sometimes it would not start at lower temperatures…thus I decreased it slightly.

The transformer was bifilar-wound on a Micrometal's T30-26 core. Fully wound it gave about 8.5 µH, which is also a little on the low side.

The input current and light output were measured and utilized as a baseline.



This is the modified schematic. The mods were arranged in 3 groups:

1) Output (1N5817) schottky diode and 1 µF ceramic cap
2) Base cap, 0.01 µF ceramic .
3) Improved transformer, wound on a T50-8 core; turns were adjusted to provide 8.5 µH

For each circuit variation, both the input current equivalent LED currents were measured and compared to the baseline, with the following formula, to provide a standardized **figure of merit as compared to the original circuit**.

$$FM = \frac{\left(INPUT_{orig} / INPUT_{\mathrm{mod}}\right)}{\left(LED_{orig} / LED_{\mathrm{mod}}\right)}$$

Circuit type	**Sensor Ohm**	**Input mA**	**Equivalent LED mA**	**Figure of Merit**
Original	231	50.6	6.7	1.00
add output diode + cap	186	52.2	11.5	1.66
add base cap	210	34.4	8.3	1.82
T50-8 core, no base cap	172	49.6	14.0	2.13
T50-8 core + base cap	172	44.2	14.0	2.39

Comments on the results:

1- Adding the output diode plus cap increases light output so much, that it is a no-brainer, always include it.

2- The base cap further improves efficiency, by lowering the AC impedance that the upper portion of the base winding, allowing to work more efficiently and turning off the transistor faster. Though in this experiment the efficiency gain was modest, I'm pretty sure that it is transistor type dependent, and other transistors may display higher improvements.

3- The transformer itself proves to be the largest contributor, no surprises here. Although the joule thief works with almost any magnetic material, for maximum efficiency it pays to use magnetic cores optimized for switchmode supply usage.

Conclusion: Clearly the joule thief, simple and fail proof as it is, has plenty of room for improvement. Hopefully in the future I'll have time to verify different NPN transistors or other circuit enhancements, which I'm sure will yield further improvements.