

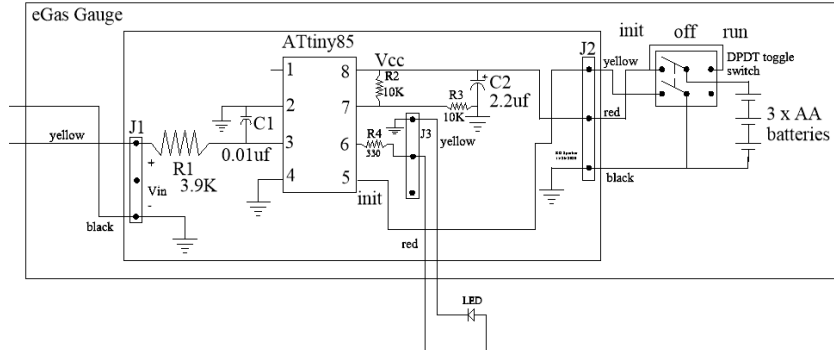
The Cry of All ATtiny85 users: I Need One More Pin! Version 1.0

By **R. G. Sparber**

Protected by Creative Commons.¹

The ATtiny85² is a complete system-on-a-chip that includes a processor, dynamic memory, Electrically Erasable Programmable Read Only Memory (EEPROM), a programmable voltage amplifier, and an analog to digital converter. All of this is inside a package with only five General Purpose Input Output (GPIO) pins. Each of these pins can be defined as an input or an output. If set to output, it can be high or low.

My latest project started out using all of these pins.



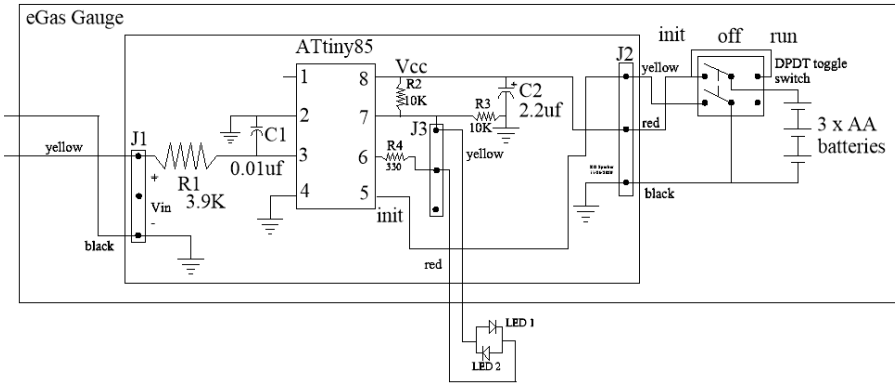
Pin 1 is Reset, and I didn't want to give that function up. Pins 2 and 3 are analog inputs. Pin 4 is ground. Pin 5 is an input, Pin 6 is an output that drives an LED. Pin 7 is another analog input, and pin 8 is power. It all fits nicely.

I was happy until I decided I wanted to drive a second LED. Sure, I could move up to the ATtiny84 with plenty of GPIO pins, but how can I better use what I have?

¹ This work is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/> or send a letter to Creative Commons, PO Box 1866, Mountain View, CA 94042, USA.

² Use the search parameters "ATtiny85 spec sheet Microchip" to see the full 221 page document.

The key was to think about how I was using each pin. I use pin 7 to measure half of the battery voltage, but that is momentarily done at startup when I don't need to light an LED.



I connected LED1 and LED2 back-to-back. One end connected to pin 6 by way of R4. The other end is connected to pin 7.

When I want to measure half of the battery voltage, I define pins 6 and 7 as inputs. This effectively makes the LEDs vanish since no significant current will flow through them.

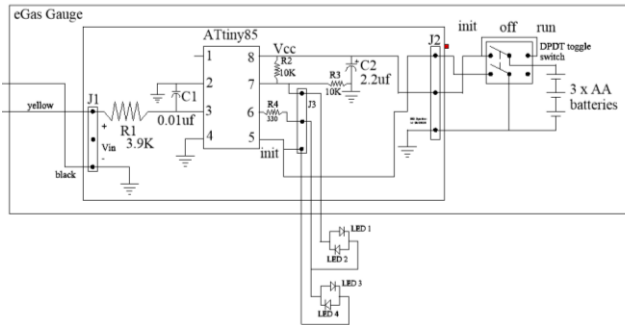
To light the LEDs, I redefine pins 6 and 7 as outputs. Then, R2 and R3 have no noticeable effect. When pin 7 is high, and pin 6 is low, LED1 lights. By making pin 7 low and pin 6 high, LED2 lights. And by switching back and forth between these states quickly³, it looks like both LEDs are on.

I dodged a bullet – no need to upgrade the device.

Over time, I decided that having four LEDs would be nice.

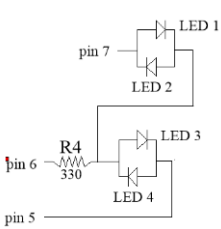
The signal on pin 5 is only used during startup. Can I reuse it?

³ I found that 20 milliseconds on time per LED is sufficient.



As wired, pin 5 is grounded during Initialization and left open the rest of the time.

If I connected another pair of back-to-back LEDs between pin 5 and R4, I could control them when *not* in Initialization.

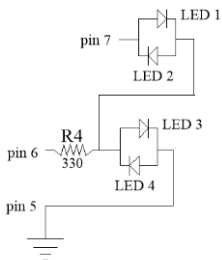


I could turn LED1 on by setting pin 7 high, pin 6 low, and pin 5 as an input.

LED2 comes on when I have pin 7 low, pin 6 high, and pin 5 as an input.

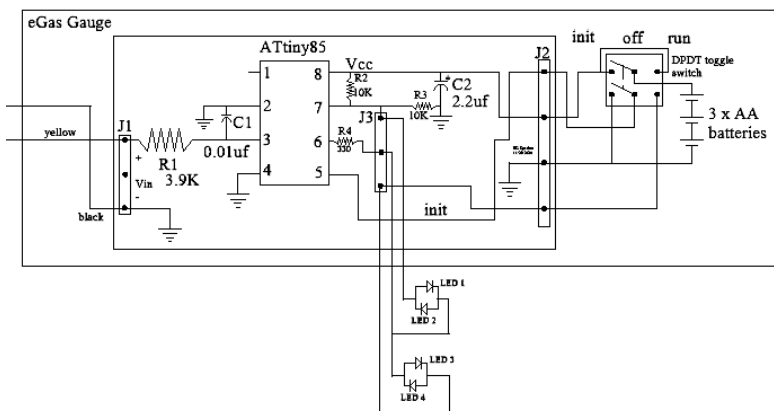
LED3 lights when pin 7 is an input, pin 6 is high, and pin 5 is low. And, finally, LED4 lights when pin 7 is an input, pin 6 is low, and pin 5 is high.

I hit a snag when in Initialization: pin 5 will be at ground via the switch.



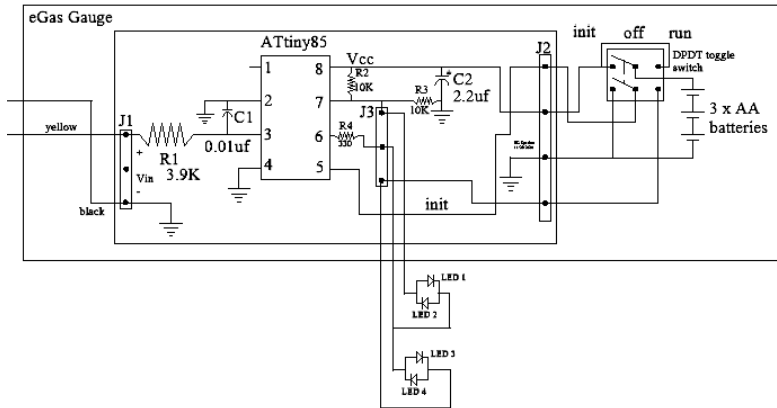
When pin 6 is high, I can turn on LED3 if pin 7 is an input. If pin 7 is low, LED2 and LED 4 would be in parallel. There would be no way to control how the current flowing in R4 would split between them. If pin 7 was high, LED1 would be in series with LED3. This would likely put pin 7 in overload. Not good.

Therefore, during Initialization, I can only reliably turn LED1 on and off. Not good enough.



All is not lost. I can rewire the switch. When in Initialization, pin 5 still sees ground, but now LEDs 3 and 4 are not connected. This lets me control LED1 and LED2, which does meet my needs.

The lesson here is to look at the entire circuit while trying to free up pins.



I now have independent control of LED1 and 2 at all times. When not in Initialization, I have independent control of all four LEDs.

It is funny how things turn out. After adding the second set of LEDs, I found a way to convey almost the same information using just two LEDs. As much as I liked being able to drive four LEDs, it wasn't needed.

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