
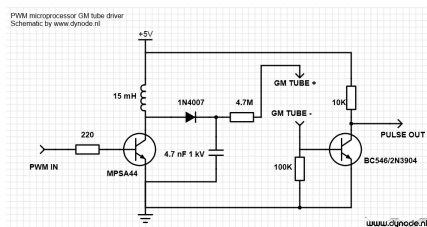


dynode.nl (<https://www.dynode.nl/>)

Trinket powered geiger counter

Posted on April 15, 2014 8:22 pm 

Lately I have been messing around a bit with microprocessor powered geiger counters. One smart guy [came up with the idea](http://theiophage.blogspot.nl/2012/09/simple-gm-tube-interfacing-to-arduino.html) (<http://theiophage.blogspot.nl/2012/09/simple-gm-tube-interfacing-to-arduino.html>) of generating high voltage using PWM signals from the microprocessor itself. With some additional external parts a HV supply and negative going pulse suitable for microprocessors is easy to make. Here is a schematic I came up with:



(<https://www.dynode.nl/wp-content/uploads/2014/04/V9epC.jpg>)

The circuit works as follows: A ~1 KHz squarewave turns the MPSA44 high voltage transistor on and off, generating high voltage when the inductor's current is shut off. The voltage depends on the pulse width of the square wave which can be tweaked in software. The 1N4007 diode rectifies this voltage, and the HV cap removes most of the ripple on this voltage. The resistor limits current to the GM tube. The current pulses from the tube generate a voltage drop over the 100K resistor which turns on the BC546. When this happens the voltage through the 10K resistor is pulled to ground, generating a negative going pulse each time the GM tube detects an ionizing ray or particle.

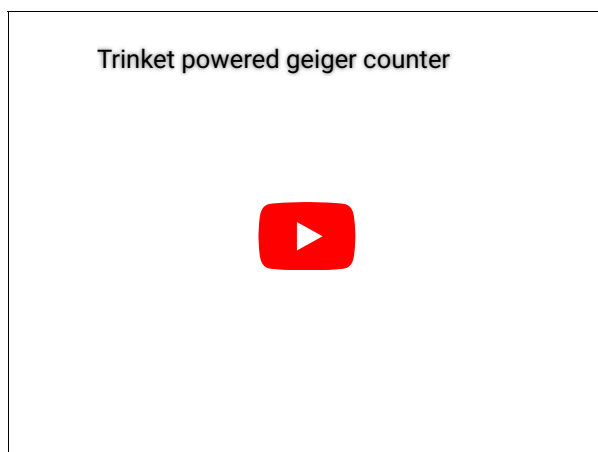
To drive this circuit I used my new [Adafruit Trinket](https://learn.adafruit.com/introducing-trinket/introduction) (<https://learn.adafruit.com/introducing-trinket/introduction>), a small board with a Attiny85 microprocessor. Using the [tutorials](https://learn.adafruit.com/introducing-trinket/setting-up-with-arduino-ide) (<https://learn.adafruit.com/introducing-trinket/setting-up-with-arduino-ide>) on the Adafruit website it is easy to work with from the Arduino environment. Here is the code:

```
void setup() {
  analogWrite(0, 30); //starts PWM on pin 0, generates about 400V
  analogWrite(1, 255); // needed to get LED to full brightness
  attachInterrupt(0, countPulse, FALLING); // attach interrupt to pin 2
}

void loop() {
  //nothing much really
}

void countPulse(){
  //pulse led
  digitalWrite(1, HIGH);
  delay(100);
  digitalWrite(1, LOW);
}
```

And here is a video of the setup in use:



Of course it is rather wasteful to only use the microprocessor to generate PWM and flash a LED. I plan on implementing counting and serial output in software later. Unfortunately the Trinket does not have native serial USB capability but bit banging a serial signal on one of the pins should work fine according to several sites. Then it is just a matter of adding a cheap PL2303 serial to USB adapter.

Update 18/4/2014

Added serial logging capability. Using a tx only software serial library (<http://forum.arduino.cc/index.php?topic=112013.0>), the Trinket outputs the measurements in CPM each 10 seconds on pin 4. New code:

```
// Trinket GM counter by Johan/dynode.nl

//counting vars
long count = 0;
long countPerMinute = 0;

// init softserial only tx on pin 4
#include <SendOnlySoftwareSerial.h>
SendOnlySoftwareSerial mySerial (4);

void setup() {
  mySerial.begin(9600); // init serial 9k6
  analogWrite(0, 30); //starts PWM on pin 0, generates about 400V
  analogWrite(1, 255); // needed to get LED to full brightness
  attachInterrupt(0, countPulse, FALLING); // attach interrupt to pin 2
  mySerial.println ("Trinket GM counter starting...");
}

void loop() {
  delay(10000); //the count is incrementing during this delay
  countPerMinute = 6 *count;
  mySerial.println (countPerMinute);
  count=0; //reset the count
}

void countPulse(){
  count++;
  //pulse led when count is increased
  digitalWrite(1, HIGH);
  delay(100);
  digitalWrite(1, LOW);
}
```

Example serial output using cheap eBay USB<>TTL serial adapter:

```
Trinket GM counter starting...
84
12
6
0
402      <--- thorium bearing mantle held next to GM tube
996
1218
1146
1074
1104
```

There still need to be some tweaking done, the circuit is quite susceptible to electromagnetic interference which causes erroneous counts.

Related Posts

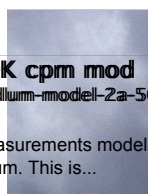


April 18, 2013 Johan

Ludlum Model 2A 500K cpm mod

(<https://www.dynode.nl/ludlum-model-2a-500k-cpm-mod/>)

I recently bought a Ludlum Measurements model 2A survey meter on eBay from Obtanium. This is...



March 20, 2010 Johan

Review: EagleTac P20A2 Mk. 2 flashlight

(<https://www.dynode.nl/review-eagletac-p20a2-mk-2-flashlight/>)

Intro The EagleTac P20A2 Mk. 2 is a LED flashlight that runs on 2 AA batteries. I bought it because...

Spits en Metro, promotors van angst voor radiostraling

Most awesome airplanes!

4 Responses to Trinket powered geiger counter



Manishi Barosee

April 19, 2014 at 2:29 pm

This is seriously impressive. I like the way how you simplified the code, great stuff!

Pingback: This geiger counter is powered by Adafruit & Atmel | Bits & Pieces from the Embedded Design World



CBGoodBuddy

June 16, 2014 at 11:46 pm

Huh, I didn't think `delay()` worked inside of an interrupt handler. `attachInterrupt` documentation says, "Inside the attached function, `delay()` won't work and the value returned by `millis()` will not increment." So kudos to you for defying convention. 😊



jens

June 17, 2014 at 9:44 am

HERE is a fucntion wqhich – as name states – eats time

```
char k_eat_time(unsigned int eatTime)
76 {
77 unsigned long l;
78 // tested on uno for 5 msec and 500 msec
79 // quants in milli seconds
80 l = eatTime;
81 l *=1323;
82 while (l-) {
83 asm("nop \n\t nop \n\t nop \n\t nop \n\t nop \n\t nop");
84 }
85 }
```