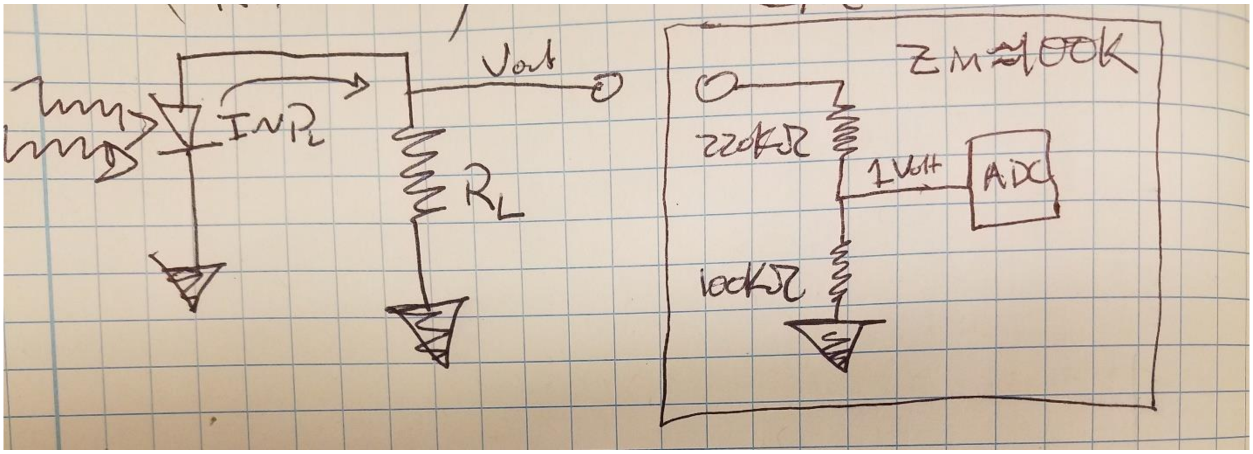
Power Meter Gizmo: Peter Kress

Schematic:



The goal is to make a power meter calibrated for Rb d1 (approximately 795nm) light between 0 and 11 mW that will assist the arranging of optical components. We used an esp8266 microcontroller with an analog to digital voltage reader with a photodiode to capture voltage produced from the incident light. I used Lualoader to transfer my .lua files to the gizmo.

We converted the measured voltage into power using the relation $P = \frac{I}{r} = \frac{V}{Rr}$ where r is the responsivity of the photodiode. After ensuring the gizmo properly reported source voltages, we used the photodiode to provide the voltage to the gizmo from RBd1 light ranging in power from 0 to 11 mW. We used two different resistor values in the voltage divider to collect voltage points in conjunction with a power meter in order to calculate the responsivity of the photodiode and convert voltage readings to power readings internally.

Using the 680 Ohm resistor led to the saturation voltage in the circuit at low power levels near 1mW. As a result, we used a 60 Ohm resistor which allowed power measurements up through 11mW. This is from the breakdown voltage, about 350mV, being reached quickly if a large resistor is used. The gizmo is designed so that any resistor may be chosen for measurements: place in the first and last spaces of the black connector connected by wires to the esp8266. If the resistor value is put into the "config.lua" code, the gizmo will calculate and display the maximum power levels it will be able to effectively measure.

Using the data points from the 60 Ohm resistor, and the fit equation $V = Vo + PRr$ we estimated the responsivity to be $r = .4761$. Rearranging the above equation, we get $P = \frac{Vin-Vo}{Rr}$.

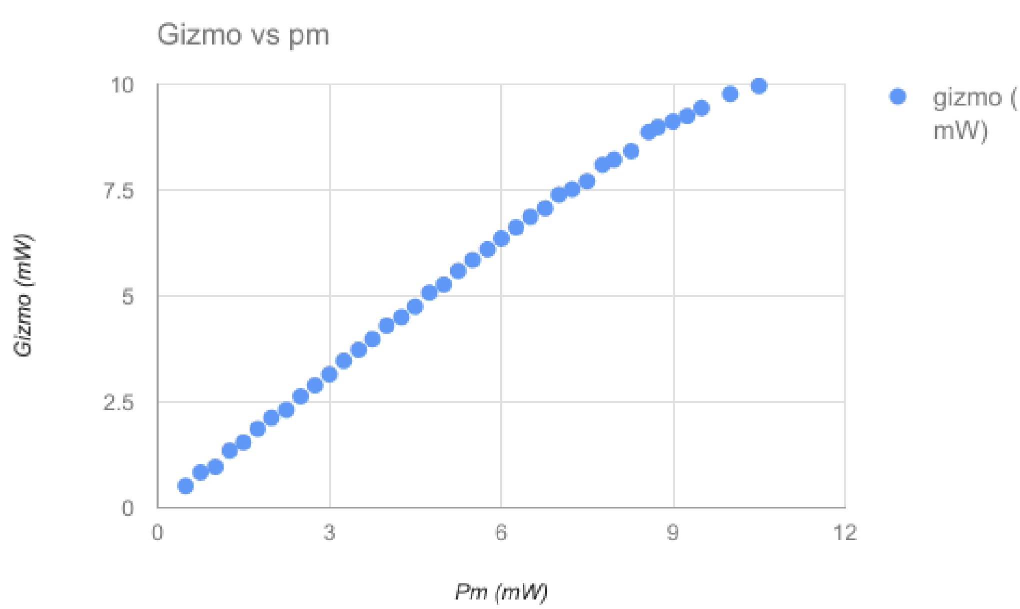
Through additional tests and adjustments on the power meter we found $P = Po + Pslope * Vadc$, where Vadc is the original voltage measured by the esp8266. $Po = 1000(intercept - (\frac{slope(calibrateintercept+Vo)}{Rr})$ and $Pslope = 1000 * slope * calibrateslope * \frac{vadjust}{Rr}$ where calibrateintercept and calibrateslope are from the initial voltage calibration, Vo is from the initial power calibration, intercept and slope are from the final power calibration, vadjust changes vadc to base 10 and accounts for the internal voltage divider of the gizmo, R is the chosen resistor value, and r is the responsivity of the diode.

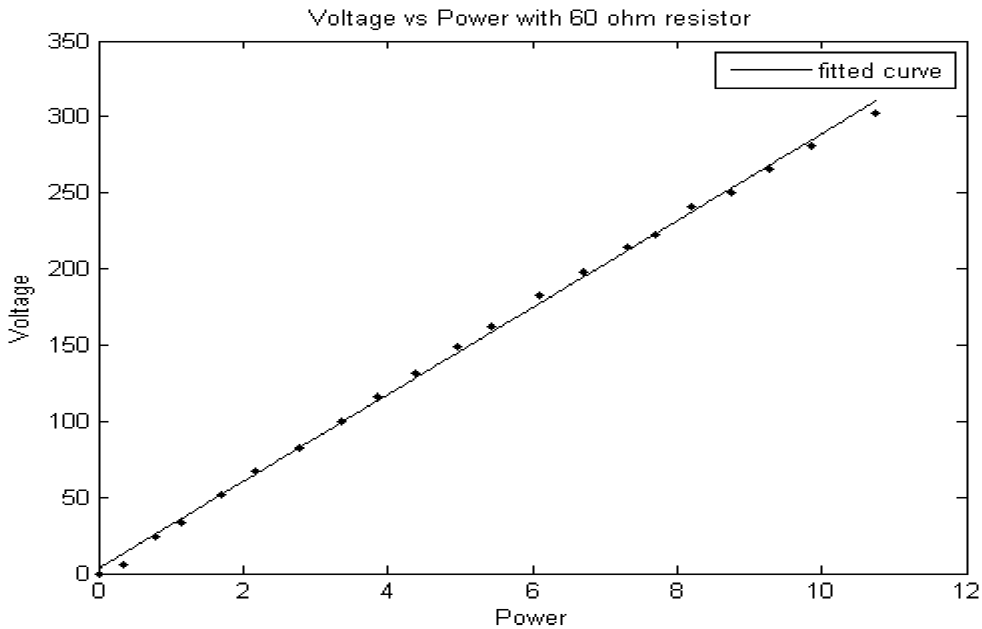
This messy set of calculations yields an easy to read power meter that reports not only the incoming power in micro-watts, but also the load resistor value, the maximum measurable power, and the precision of the power meter. The max power and precision are determined as a function of the resistor value to prevent reaching the breakdown voltage of the diode which is 350mV.

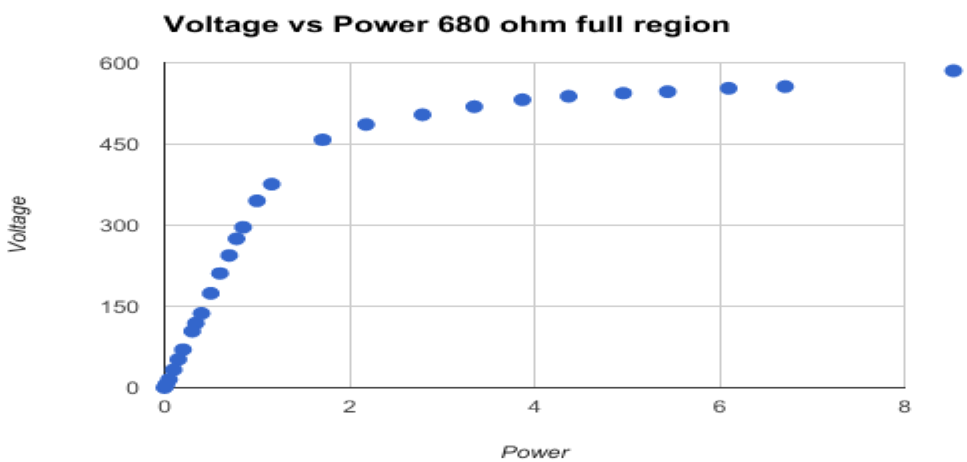
The Resistor value can be altered on line 47 of “config.lua” in \\qo.physics.wm.edu\qol_grp_data\Peter\VoltageStation\Lua files\config.lua. None of the other parameters need to be changed. Ideally, an ohmmeter could be coded into the power meter so it could automatically detect resistor values without any code adjustment or re-flashing the device.

The power meter requires an 8-32 tap screw to bolt the photodiode component into an optics table stand. The power meter requires power from a usb cable as well, and connects to the photodiode via a connecting cable across 3-pin connectors on both components.

The first figure shows the final calibration data between a thor labs power meter and the gizmo which was used to create the last adjustments on power calculation. The second figure shows the estimation line for the responsivity of the diode. The third figure shows the failure of the power meter as the voltage approaches and surpasses the breakdown voltage of the photodiode.




Voltage vs Power with 60 ohm resistor
fitted curve
Voltage
Power
0
50
100
150
200
250
300
350
0
2
4
6
8
10
12


Voltage vs Power 680 ohm full region
Voltage
Power
0
150
300
450
600
0
2
4
6
8