Project goal complete, ...

To the left is the High Sensitivity Knock Switch Circuit, a long name but effective circuit. Now all I need is to add the charger but it can’t connect to the Arduino directly.

The 5V rail connects to the COM pin and bridged to the NC pin normally, but switches to NO pin when a vibration is detected. Disconnect the COM, NC, and NO pins and connect the USB phone charger cable instead. Before that can be done the USB cable must be spliced and have the internal positive aka the red wire cut to reveal 2 exposed leads. If the USB cable is long, so the 2 new leads are capable of reaching the breadboard without trouble, but if not, then a long enough wire must be soldered to each exposed lead.

Connect one open charger lead to the COM pin and the remaining open charger lead to the NC pin, while leaving the NO pin disconnected or connecting to the charger ground.

Extending the Project

Viola! The USB cable can now connect the charger to the phone’s charging port. If desired the EN pin can be connected to GND pin to disable the charger power, or even better adding a manual switch or relay.

This project has not done yet though. There’s more that I wanted to share, but didn’t have space for or time to pursue.

I’d like to thank my parents and big brother, Professor Vigs, my NET 499 class and all the previous classes. I want to extend a special thanks to David Houlding, everyone at Arduino, Adafruit, Vilros, Amazon, Youtube, Instructables, and for all the unnamed people who share their work for someone else to pick up on and make something better.

Battery Charger Automation

By William Gordon
NET 499 Senior Capstone Project
Spring 2015
**Altoids Tin Phone Charger**

Parts: PowerBoost 1000 kit w/USB connector and 1x Terminal Block, 2x Rechargeable AA batteries, 1x2 AA Battery Holder w/battery leads, 2x Cut Velcro w/Adhesive, USB Charging Cable, Phone, and of course an Altoids Tin w/USB opening

**Knock Circuit**

Parts: Arduino Uno R3, 1x USB A/B Cable, 1x 1M-Ohm Resistor, 1x Piezo Disc, 2x 22-Gauge AWG Wires, and 1x Breadboard w/Rails

Download and install Arduino app for PC and find Knock program in app or arduino.cc. Detects vibrations and determines if loud enough to be a “knock.”

Benefits: simple beginner circuit, no soldering, safe 5V power, can get parts needed besides the 1M-Ohm resistor from the Vilros Ultimate Starter Kit

**Knock Triggered Relay Switch Circuit**

On the graph paper below, I drew combined a circuit from the Vilros Ultimate Starter Kit Guide booklet Circuit #11 with the Knock circuit can be found at arduino.cc or download the Arduino app, then I edited the photo to make it more visible to read. The Vilros Ultimate Starter Kit includes all of the parts for this circuit

Parts: all the parts included in Knock Circuit, 2x 330-Ohm resistors, P2N2222AG NPN transistor, 1x 5V relay, 1x 1N4148 Diode, 2x LED’s, (phone charger option explained for next circuit)

Download and install USK code from vilros.com. Use example #11 program now accessible in the Arduino app.

Benefits: beginner to intermediate level circuit, small design, safe 5V power, can get the parts needed besides the 1M-Ohm resistor from the Vilros Ultimate Starter Kit

**Inspired to add on for the High Sensitivity Knock Switch Circuit**

With the issue of not having a reliable signal change from threshold, delay values, or simply the vibration is too weak. There’s more to sensing a piezo signal. David Houlding’s High Sensitivity Vibration Sensor design (at http://davidhoulding.blogspot.com/2014/02/high-sensitivity-vibration-sensor-using.html), does just that.

Parts: All parts included in Knock Triggered Relay Switch Circuit, LM358N Operational Amplifier Chip, 1x5V Zener Diode, 1x 100nF Capacitor, 1x 100K Ohm Resistor, 1x 100K-Ohm Resistor, 1x 1K-Ohm Resistor, 100K-Ohm Potentiometer,

With adding the Knock Triggered Relay Switch Circuit to my own, it would replace the Knock Circuit for a better piezo signal. The sensitivity can now be adjusted on the board using the potentiometer and voltage input ranges from 0V to 5V since the Arduino max inputs shouldn’t exceed 5V.

Shown Below is this circuit I designed with the Fritzing App. I added a battery charger and a LiPo battery to represent the Altoids Tin Phone charger and a phone being charged, since options were limited in that respect.