

Hackaday Fubarino Contest Entry

Project: Bluetooth PC Remote Power Switch (Prototype)

Description:

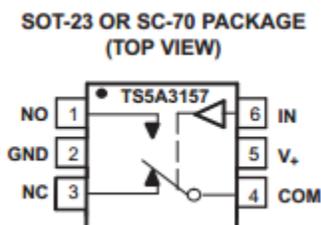
This projects goal is to create a remote power switch for a computer. Instead of using WoL or other remote start protocols, I wanted to design a device that interfaced with computers hardware (eg. The pin headers on the motherboard that connect to the power and reset switch).

Demo video at the end of the document (Youtube link <http://www.youtube.com/watch?v=s08A1dteivM>) This shows the easter egg (more details at the end of the doc)

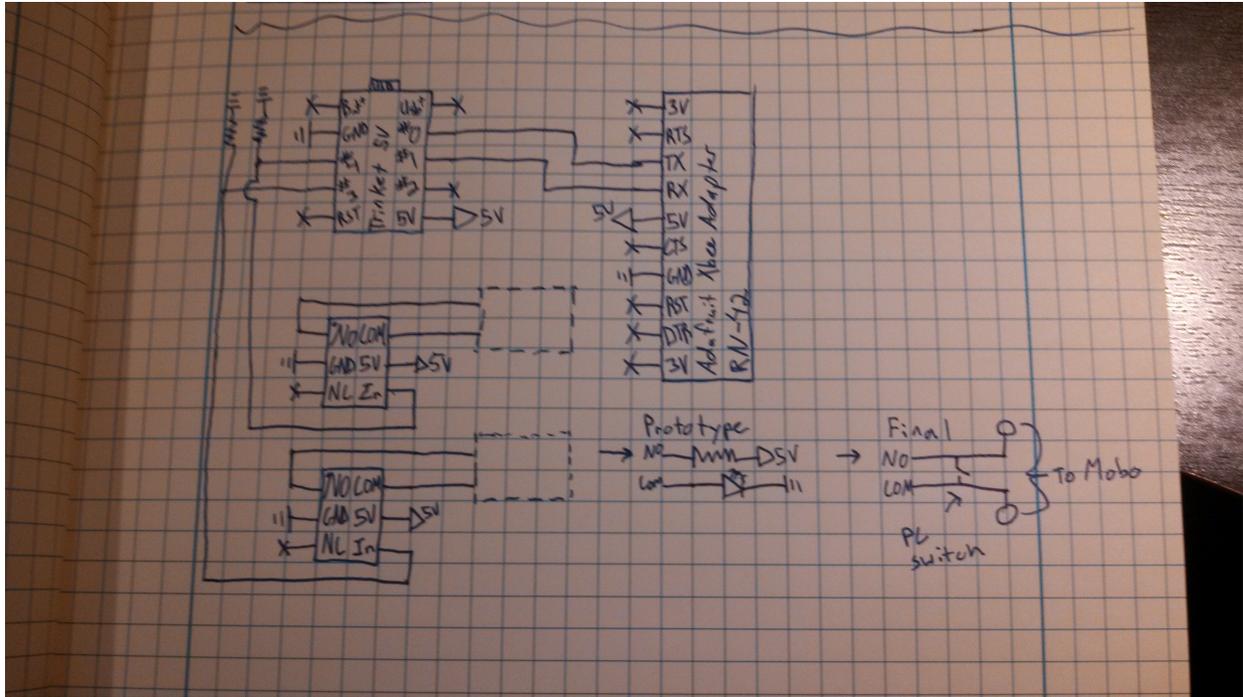
Be sure to look at the future plans for this project!

Design (Hardware / MCU):

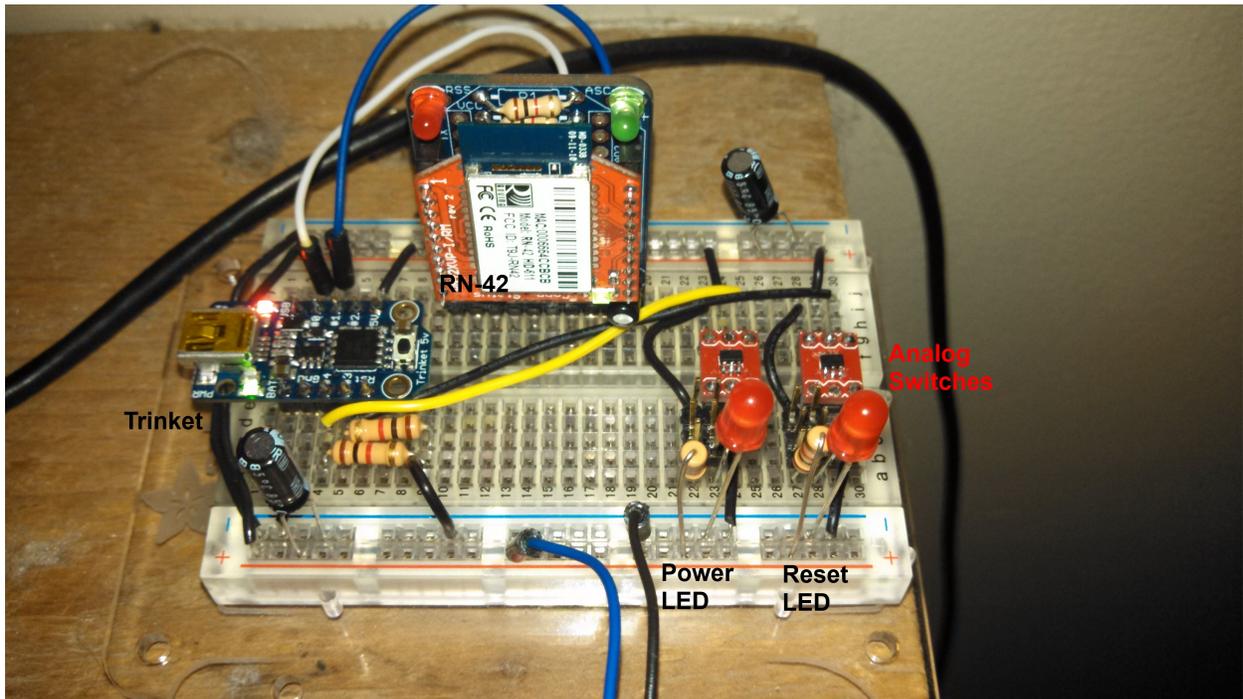
The first revision of the hardware design for this project consists of the Trinket that I received from the Hackaday Trinket Contest. (Awesome little development board by the way! Thank You!) The Trinket was programmed with the Arduino IDE (primarily for fast development) and was uploaded over USB. This process was simplified with the IDE download from Adafruit. The Trinket is connected to a RN-42 Bluetooth module (with voltage regulated from an Adafruit xbee breakout board) via Software Serial (Baud 9600), and two TI Analog Switch Ic's (TS5A3157 - <http://www.ti.com/general/docs/lit/getliterature.tsp?literatureNumber=scds219a>). The analog switches are connected to red LEDs in the current revision of the hardware. This allows for testing without embedding the device inside of a computer. All of this hardware is powered by 5V. The 5V Standby power line of a standard ATX power supply will be used to power the hardware in the next revision of the hardware so the device can be completely embedded inside of a desktop computer. One issue that I currently have is with the analog switches only allowing current to flow in one direction and offers 10 Ohm of resistance. I do not know how this will perform when I get it connected to a PC. I can always replace them with relays. (Pictures Below)



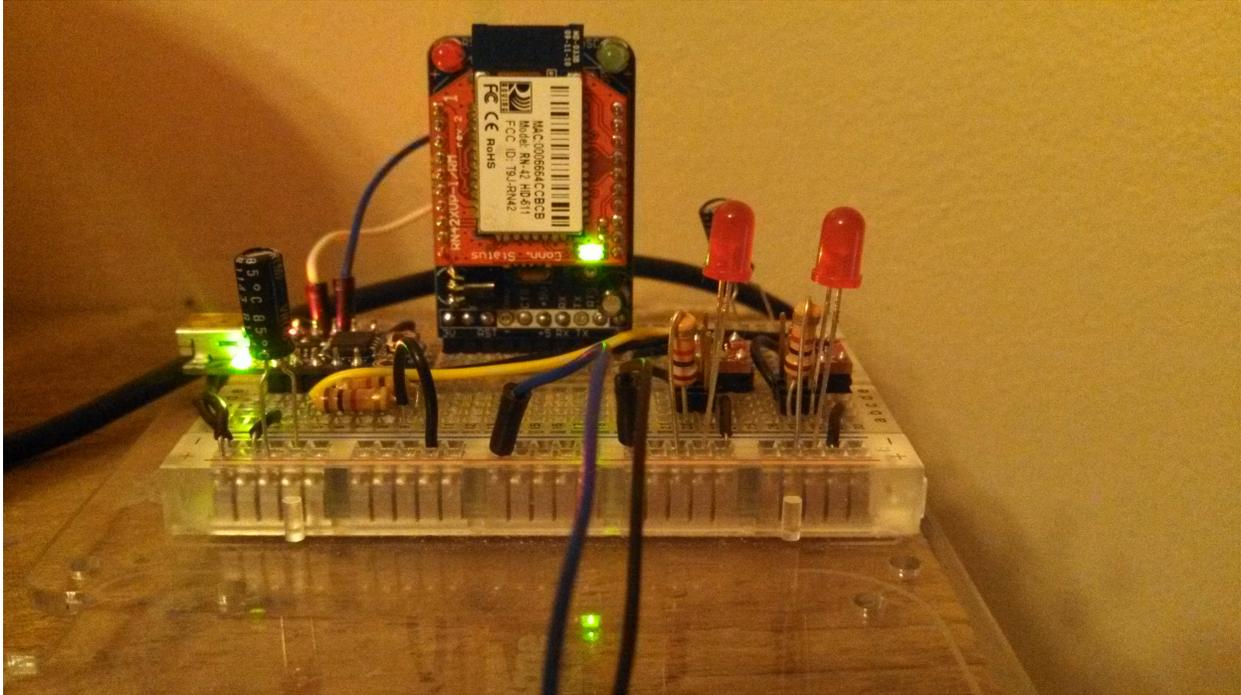
TI TS5A3157 Diagram



Initial Schematic (will convert to eagle at a later date)



Top view of breadboard (Hardware Revision 1)



Side View of breadboard

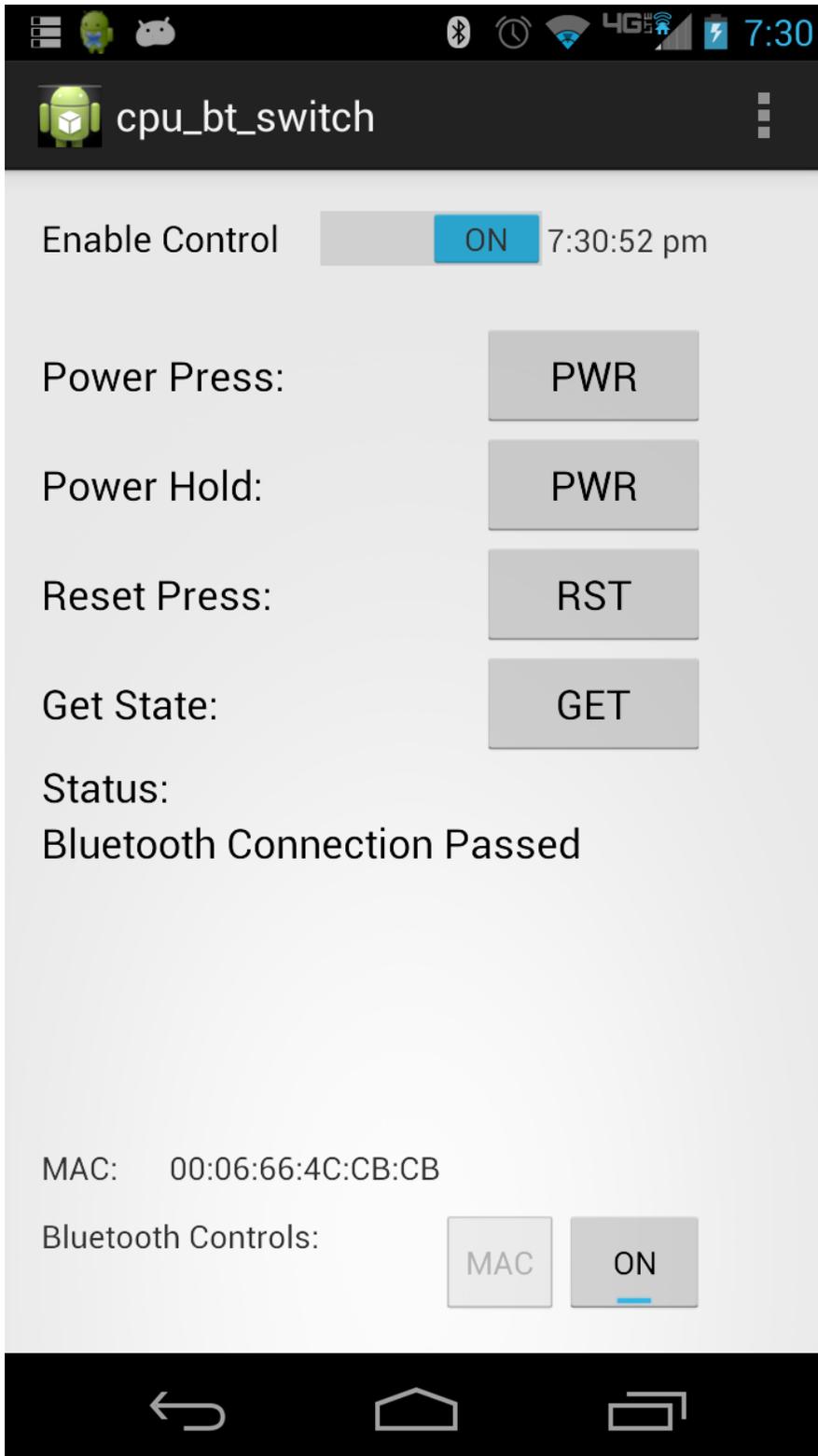
Design (Trinket / Firmware):

The firmware is very basic on the Trinket. The firmware starts the Software Serial port at a 9600 baud rate. The Trinket Firmware then enters the main loop which handles the Bluetooth messages and the analog switch signaling. Upon every command received a message is transmitted back to the sending device to tell the sender what is happening. This call back message will eventually be used to indicate the state of the computer attached to the device (via a sense line). This will be done with the single left over pin on the next revision of the hardware. **THE EASTER EGG! Every 42 power commands sent to the device the call back message will contain <http://hackaday.com>.** The signaling is handled with a millis() comparison on the Trinket allowing both power and reset to be pressed at the same time (this will be filtered out in the next revision of the firmware).

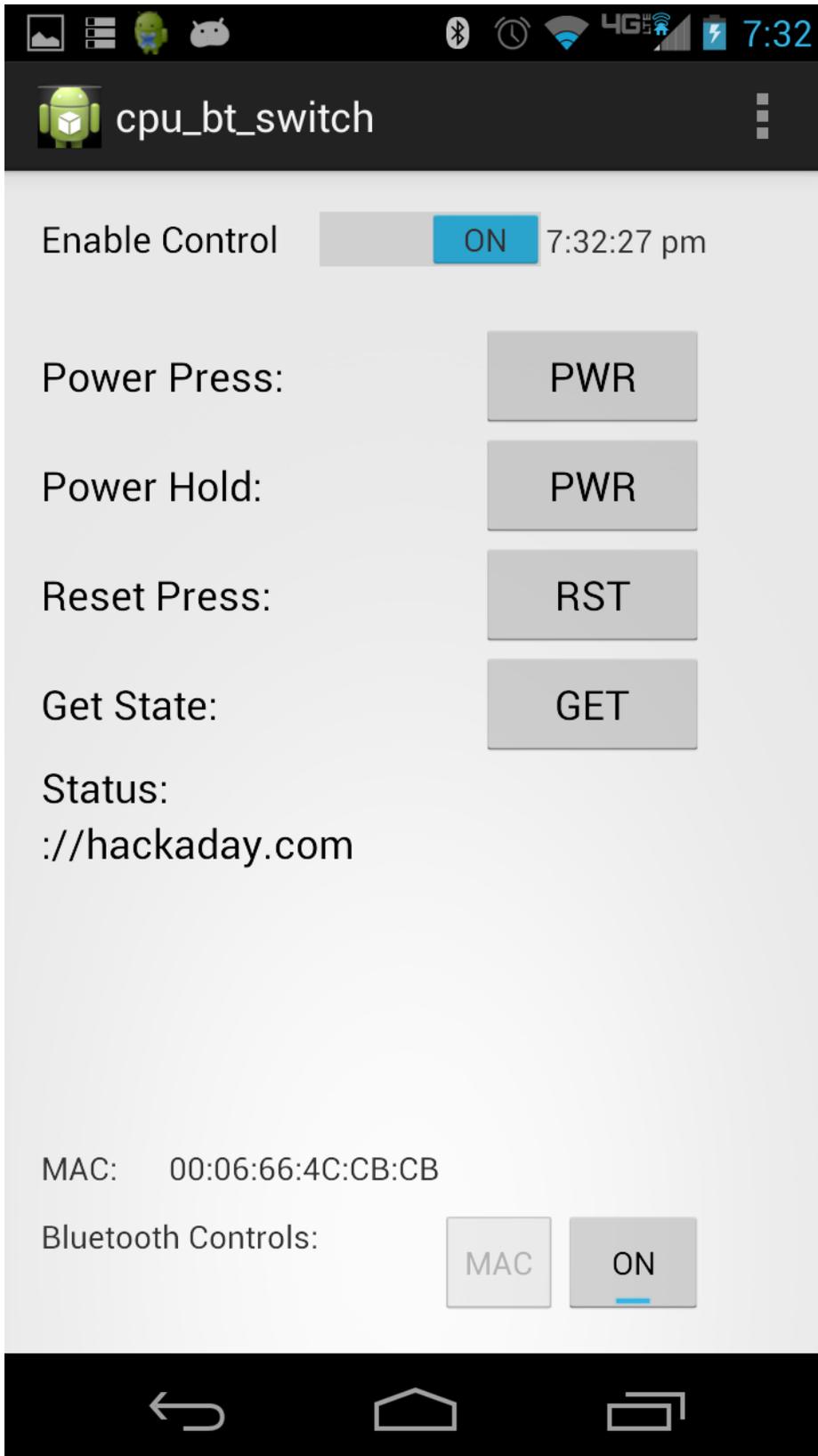
Design (Sending Device):

I felt an Android App was in order. I designed a basic Android App that connects to the device over Bluetooth. When buttons on the App are pressed, a command byte is send over the Bluetooth connection and the Trinket does some processing. There is a field in the App that displays the call back message from the device **(which can display the Hackaday Easter Egg)**. I wrote a broadcast based Bluetooth service to handle communication to the App Activity. I am having some issues with threading in the android app which is causing a display issue in the call back field (labeled "Status"). Creating this

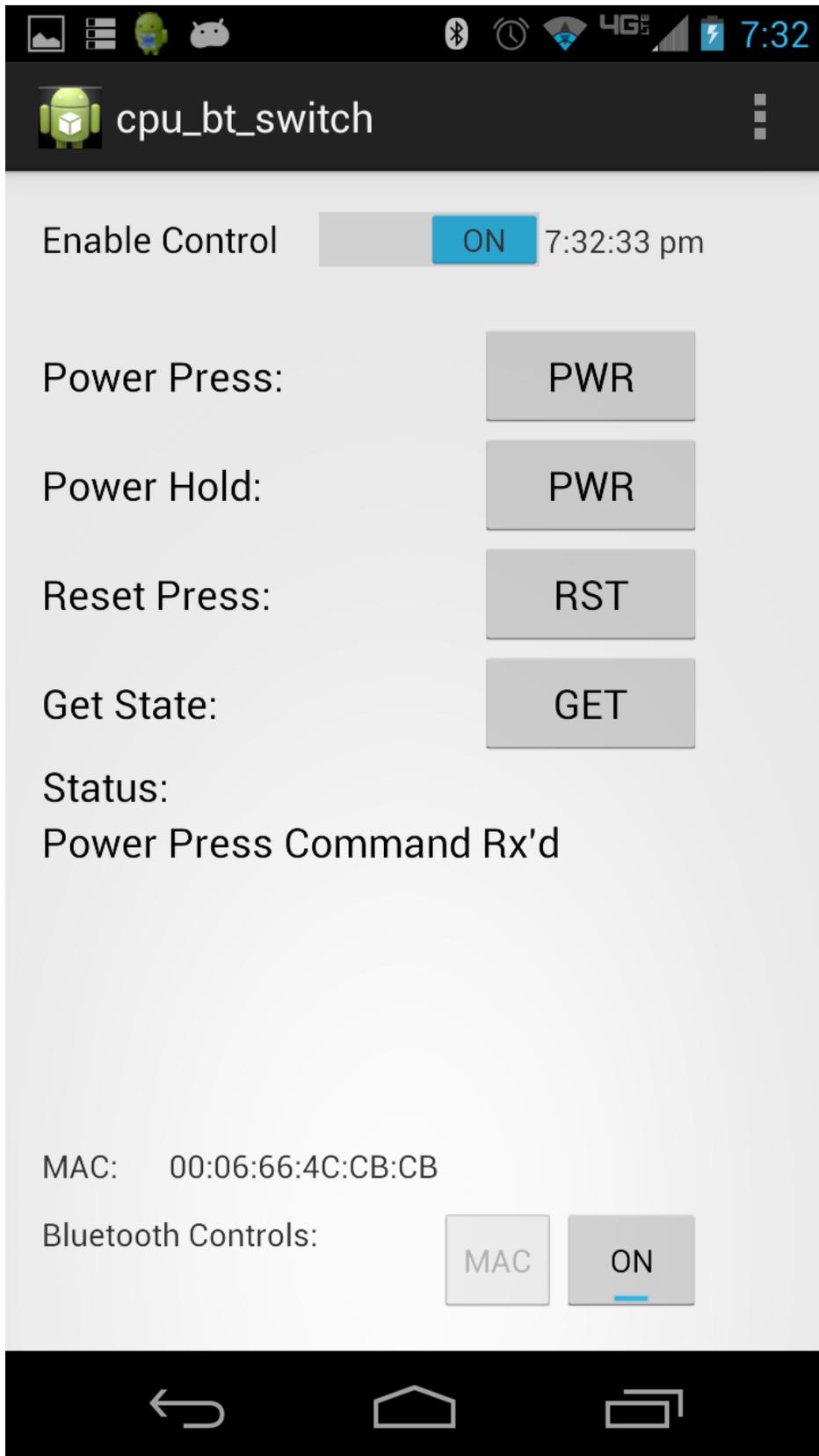
service is nice because it will allow for rapid development of other Bluetooth controlled apps in the future. It is planned to allow the changing of the MAC address of the receiving Bluetooth device, but this has not yet been implemented. (See pictures below)



Android App Showing Connection to Bluetooth module



Android App showing Hackaday Easter Egg (with display issue)



Android App showing Call back message

The DEMO Video (<http://www.youtube.com/watch?v=s08A1dteivM>) shows the app working and the easter egg. I reset the Trinket and ran 30 + power commands prior to shooting the video.

Future Project Plans:

Add additional functionality to the firmware (as described above)

Rework the hardware to use a HC-05 Bluetooth module

Create a custom PCB with breakout headers for the power and reset switches

Using the HC-05 Bluetooth module

Splice an ATX extension to get access to the 5V Standby power line, the 5V line, and Ground.

Add features to the Android App (login, scheduling, better logging etc)

Run a web server on a Raspberry Pi that interfaces to a Bluetooth module to allow complete remote access to the device (downside is the proximity of the Pi).