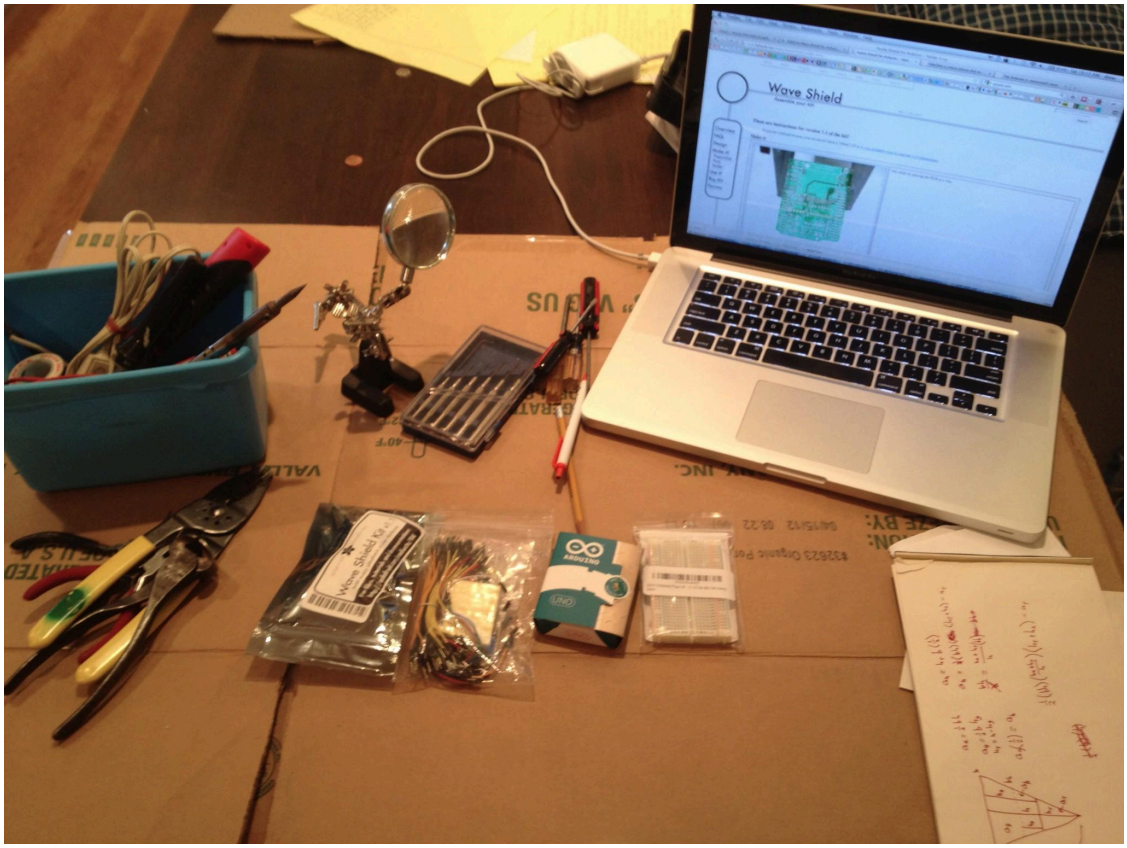
the phone that talks to god

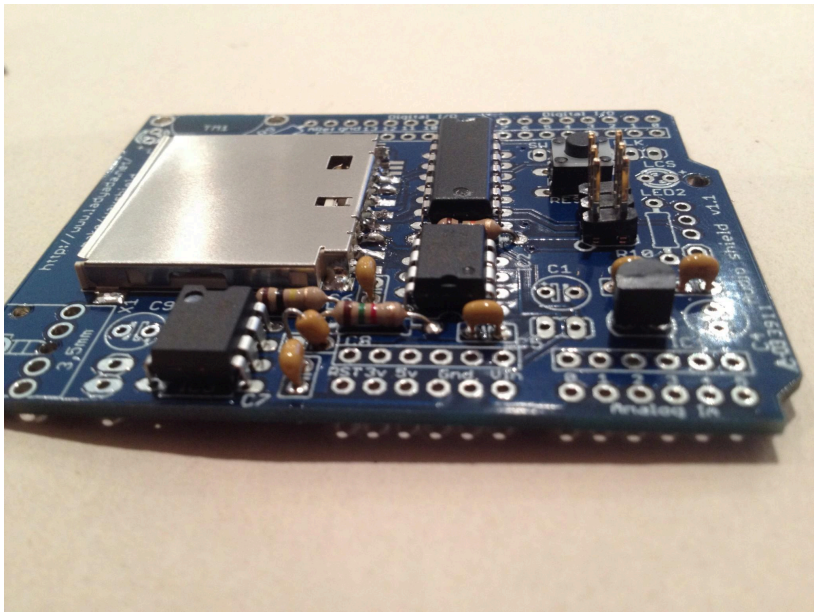
My good friend Gallagher had an idea for a phone that talks to god, which I stole (with permission) and modified to my own ends with help from my dad.

The general idea was to have a curio that people would want to check out during my shows, hopefully with the side effect of making them more likely to pick up CDs. After some research I consolidated the following:

One rotary phone (from antiques store, about 12 bucks)
One Arduino kit (http://arduino.cc/en/Main/ArduinoStarterKit)
One Adafruit Wave Shield (https://www.adafruit.com/products/94)
One father, with vastly superior soldering skills, electronic know-how, and soldering equipment
One pair of headphones for testing
One 12V power adapter
One SD card and SD card writer



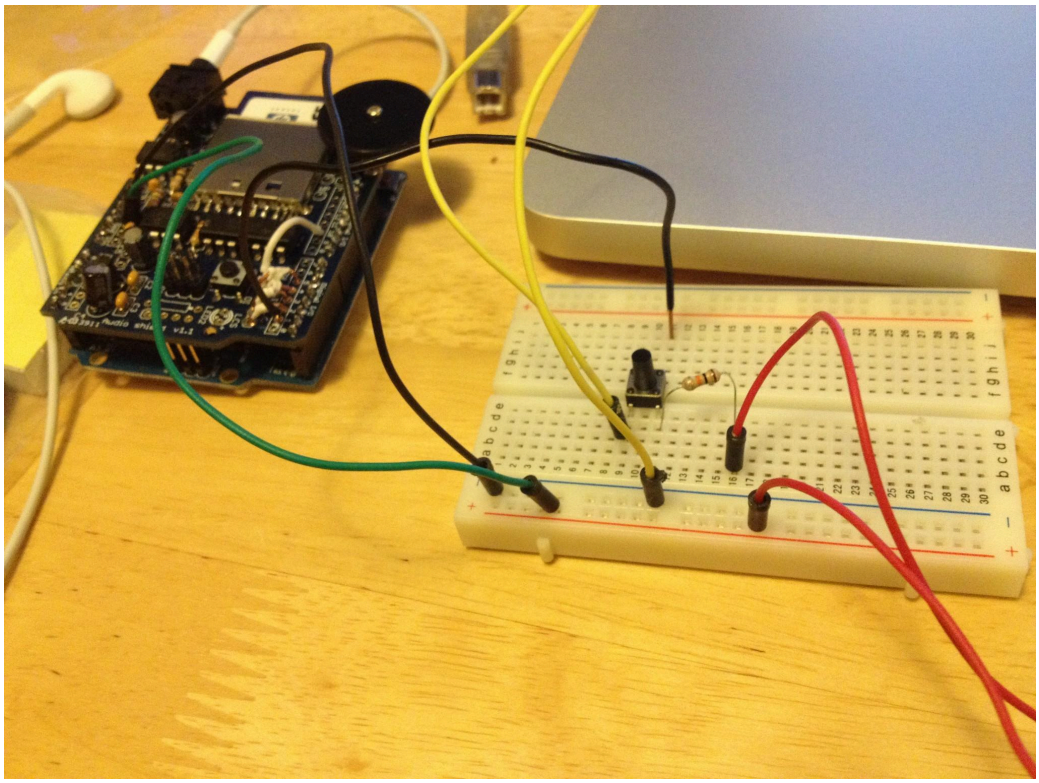
This was a great project for a novice such as myself because it just combined two pre-existing pieces, specifically the Adafruit Wave Shield sketches (http://www.ladyada.net/make/waveshield/examples.html) and the rotary input interface from instructables.com (http://www.instructables.com/id/Interface-a-rotary-phone-dial-to-an-Arduino/#step1)

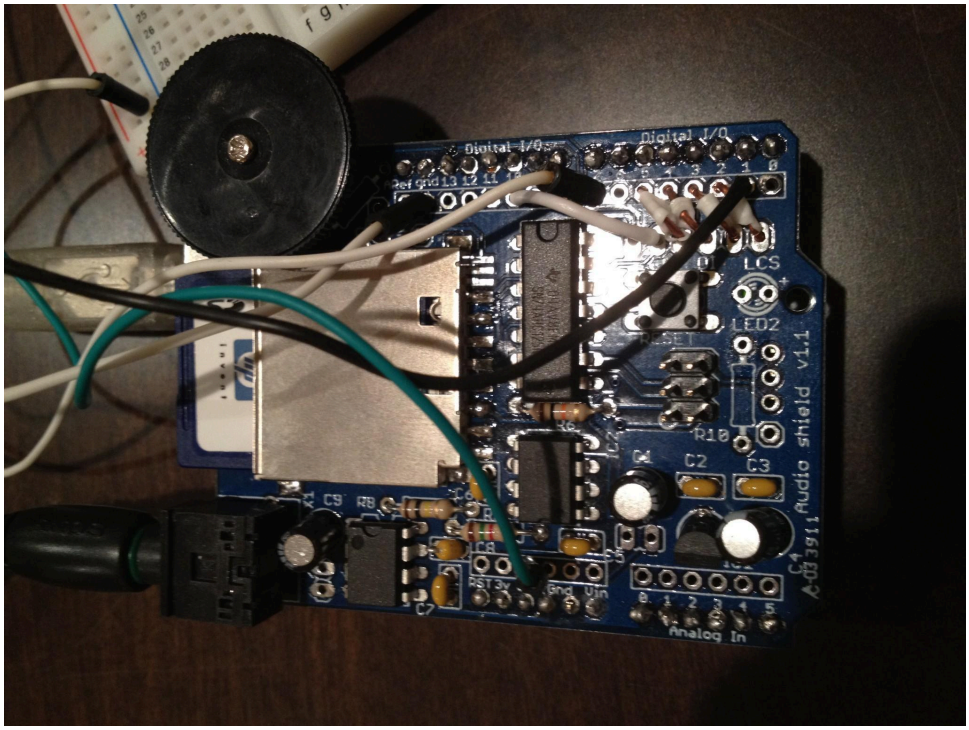


After I'd run a few beginner sketches on the Arduino, my dad and I donned our soldering boots and put together the Adafruit Wave Shield. The shield ended up working great, and the documentation and assembly instructions were clear and precise, but it was a good deal more soldering than I had anticipated. I relied heavily on the expertise of the aforementioned electronically-skilled father. If I were to do a similar project I might consider researching pre-built shields or spend a couple of weeks steeling myself for significant soldering time.

Once the shield was put together we tried out some of the Adafruit sample sketches (http://www.ladyada.net/make/waveshield/examples.html), using a sonar ping loaded on an sd card. Surprisingly enough, the shield worked the first time, a sign that our soldering work went much better than could have been expected.

Confident from our success with the waveshield, we followed the rotary input instructable (http://www.instructables.com/id/Interface-a-rotary-phone-dial-to-an-Arduino/#step1) and got the serial console to print out the correct input from the rotary phone.





At this point it was just a matter or combining the two bits of code into one, a rare task that matches my skillset better than my dad's. That code is here: (https://docs.google.com/document/pub?id=1j1TKho_30xcF9V-rxQt28pRAT3tQijxC2EsZljv5Y68). It's not the cleanest code in the world, as it's a mix of two example sketches with my own haphazard modifications, but it should provide some idea what's going on.

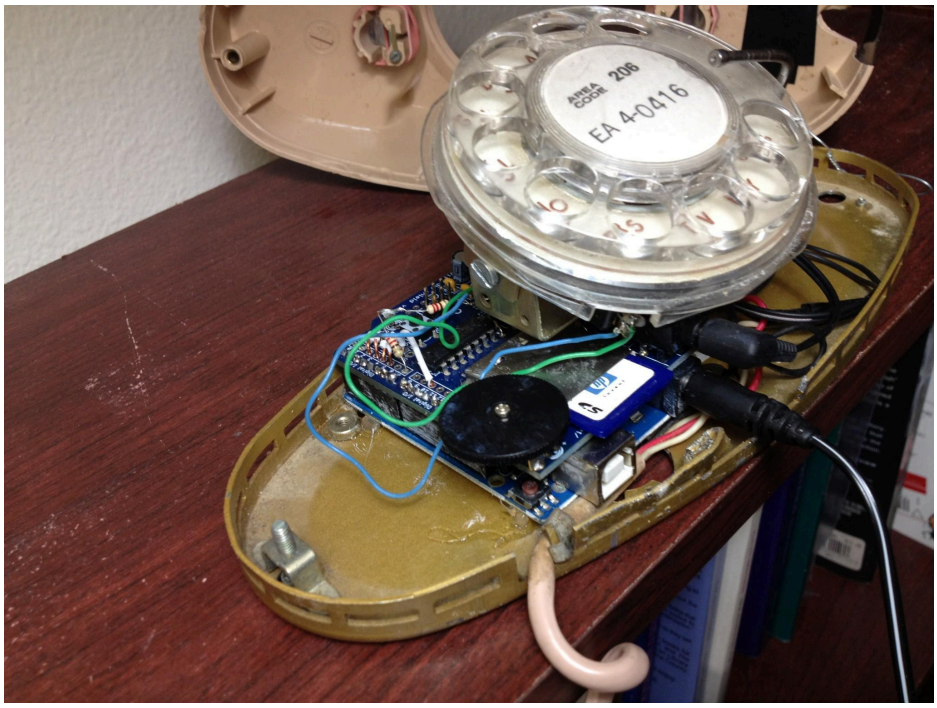
My original plan had been to buy a headphone handset, like this one (http://www.amazon.com/Native-Handset-Android-Blackberry-Samsung/dp/B003DKL4KE) but my dad was more ambitious and with a little work we were able to strip the wires from the rotary phone handset and solder them directly to the headphone jack that comes with the Wave Shield. This took a little experimentation to get right, especially since the audio wires in the handset were coated with some sort of protective material underneath the sheathing that we had to burn off with a lighter.

The SD Card I had was of limited size, so I ended up only using five "numbers" for my initial design. Here are the audio files by digit -- if you dial "1" you hear the sonar ping, etc. I usually write out a key on a piece of paper before a show and then lose it before the next one and have to write it out all over again:

1 - Sonar Ping
2 - Monster (one of my songs, creative commons licensed)
3 - God (I used an old phone error message)
4 - The Devil (I chose an excerpt from an interview with the devil)

5 - Tycho's Nose (one of my songs, which will be creative commons licensed if I ever get around to releasing it)

Once I had all five songs working we set the Arduino/Shield unit in the phone casing with some super glue and plugged it in with a power adapter. With a little work with knives and pliers and wire we were able to to get the whole thing to look pretty self contained, though I misplaced the bit of metal that stops your dialing finger at the right spot. A bent nail solved that problem.



Some issues I ran into:

-Sometimes a song would stop in playback and the whole thing would freeze. Unplugging and replugging in would usually solve this, but I never figured out what the issue was. It seemed like messing with the serial buffers would affect this, sometimes positively.

-I had trouble getting the files on the SD Card to be in a specific order... eventually I punted on that problem and just wrote the key to match whatever order they ended up in.

-Everyone expects that if you hang up a phone, it will stop making noise. I didn't hook anything up to the two little hang-up nubs on the phone body, and when the sound file keeps playing after you hang up people get confused.

It turned out pretty great, all in all, though I don't know if it's made people more likely to take CDs home. Thanks for the read, and hope you get a phone that talks to god, too!