Exploring the Ultrasonic Theremin

SensoredHacker @ The Bangor Makerspace

March 25, 2024

Abstract

The Ultrasonic Theremin is my first exploration into creating MIDI devices. The original Ultrasonic theremin was a derivative project inspired by a segment describing the evolution of aerospace controls in *Hitch Hikers Guide to the Galaxy*. The initial project sent navigational control signals to a robot using hand waving, which proved difficult to controll. So I wrote a simple piano app using SDL in C to play sounds using keystrokes instead. That was ages ago. When a member at the Bangor Makerspace asked if I could make a MIDI musical instrument, it got me thinking about this project again. Due to my lack of skill with more traditional instruments, I thought it would be neat if I could play sounds by waving my hands around like a lunatic. The Ultrasonic Theremin was born.

Premise:

The Premise of the Ultrasonic Theremin is to move around making sounds. The general operation is that within an ultrasonic field, a range of distances can be detected, and those ranges are mapped to a range of MIDI notes.

Prototyping:

Like most electronic projects, this one began on a breadboard. I put 4 HC-SR04 Ultrasonic sensors on a bread board side by side, and wired them into a Teensy 4.1. I read about the MIDI protocol on a variety of websites, and utilized some example code from the Arduino website to create a basic working model. I downloaded the first free MIDI synth software I came across, and used that to translate MIDI notes into various sounds. It worked. I duplicated the conditional statements of my code 3 more times, and wired in 3 more ultrasonic sensors. Testing it again was interesting. Given the proximity of the sensors to each other, I determined ultrasonic waves transmitted from one sensor could be detected by neighboring devices. Playing with the prototype I also noticed there is a conical field around the transmit transceiver. This makes the sensor detect objects sooner on the transmit side.
Given this new information, I was ready to build a second prototype with increased sensor spacing, and some minor code improvements.

Construction:

I found the nicest plank of wood from my scrap pile that was around 32 inches long. It has beveled edges, and is sanded. I think it came off an old futon. You may want to build a nicer platform. I created 4, 16 inch long wiring harnesses using 22AWG solid core wire, with a 4 pin female header soldered on one end of each. I glued a small breadboard to the center of the board, and placed the teensy 4.1 there. The wiring harness color layout I used was: Red, Yellow, Blue, Green, for +5VDC, Trigger, Echo, and Ground, respectively. I used hot glue to fasten the wiring harnessed to the board, with approximately 6 inch spacing. The wire side of the harness was plugged into the breadboard.

Software:

Basic overview of the code: The code continuously reads distances from four ultrasonic sensors and translates them into MIDI notes, creating a musical instrument controlled by hand movements in front of the sensors.
Include Libraries: uses MIDI.h library for communicating with MIDI devices.
Define Pins: Defines the pins for the ultrasonic sensors and sets the range limits for notes, distance, and velocity.
Setup function: configures all sensor pins as either input or output.
Loop function: reads the distance from each ultrasonic sensor, maps the distance to MIDI note values and velocity, and sends MIDI note on / off messages based on the mapped values.
getDistance function: triggers an ultrasonic sensor and measures the time for the sound wave to return, calulating distance based on measured time.

Source code for this project is available at:
https://github.com/FOSSBOSS/Ultrasonic

Utilities:

This project was coded in the Arduino IDE, on the Linx platform. Various Linux utilites for reading the MIDI datastream were used, including: amidi, and aseqdump for debugging purposes.

The Ardunio IDE is freely available, and features a wide range of supported devices, and truely excelent documentation.
https://www.arduino.cc/en/software

The synth software used for demonstration is helm.
https://tytel.org/helm
Helm was fun and easy to use, and freely available. It doesn't build out well on a Raspberry Pi, I would reccommend only trying it on standard PC devices.

Hardware:

This project was built using a Teensy 4.1 micro controller development board, available from https://www.pjrc.com/teensy This project may not explicitly require a Teensy 4.1, but if I were going to advocate buying anything, from anywhere, it would be Mirco controller development boards from www.pjrc.com. Paul, and his team provide excelent documentation, support, and products. If youre brand new to elctronic projects its probably worth it to buy from somewhere that you can get support from. This isn't sponsored, I just like what they do.

HC-SR04 Ultrasonic sensors use as many as you think you need. Maybe its 4. Maybe its more. Its a ubiquitous part at this point in time. 16 for 16$ on amazon.

Odds and ends:
a peice of scrap wood, some 22 AWG wire, female pin headers, and hot glue were also used in this project.

Caveats:

The current implementation of this project is a prototype. There is loads of room for future improvements and features. pitch bend, range control switch, or an on / off switch would all be excelent additions.
Why 4 sensors? Thats all I had around, figued just use all of them. MIDI supports up to 16 channels per device.
I tried to use descriptive names in the source code, but neither this document nor the source code are intended to teach you C/C++ if you don't allready know them.
This project was contructed in approximately one hour, and put on display at the Maine Science Festival in Bangor Maine for the following 2 days. (March 23 thru 24, 2024) An untold number (hundreds? thounsands?) of people expressed wishes to purchase this device, or garner assistance in making thier very own.
I'm down, lets do it. Contact the Bangor Makerspace on
Facebook:https://www.facebook.com/thebangormakerspace
or join our discord server and inquire. https://discord.com/invite/UZkcecgRch
We're happy to help.

Conclusion:

Conclusions are for finished projects.