



# The Manual/Story of the PharmMonster

**Scotty H**



# TABLE OF CONTENTS

<b>THE IDEA WHY?</b> .....	<b>1</b>
The name.....	2
<b>INSTRUCTIONS</b> .....	<b>5</b>
2x Buttons: .....	5
Oregon Trail game .....	5
3x Switches.....	7
8 Ball Mode .....	7
<b>THE PLATFORM HARDWARE</b> .....	<b>8</b>
Processing .....	8
Audio.....	8
<b>THE PINOUTS IO WIRING</b> .....	<b>11</b>
Pinout wiring .....	11

**WIRING .....****14****THE EASTER EGG PLANNING PROCESS.....****16****AUDIO FILES EXPORTING .....****18**

The process.....	18
MIDI settings in MAC OS x .....	19
Ableton example .....	20

**DIAGRAM FUNCTIONS .....****21**

The PharmMonster anatomy:.....	21
--------------------------------	----

**AUDIO INPUT ACCOUNTING AMBIENT NOISE .....****22**

The level .....	23
-----------------	----

# THE IDEA WHY?



The world needs more robot monsters. After talking about stuffed animal ugly monster dolls with Molli, there was a joke of creating an Ugly Doll for the pharmacy. Since all I do in my life is work on strange computer projects, the PharmacyMonster was born.

He needed to have a strange look and have the glowing personality of a superstar. The attempt to make him as adorably ugly would be made, however, it was a tall task to make something as lovably strange as the pictured DIY Ugly Doll, or an actual Ugly Doll brand stuffed animal.

The monster having the ability of sensing the world around him was a must. After bouncing the idea off individuals, I got fantastic feedback such as:

**“I’m a little confused on why...”** - random person at programming meetup

and,

**“So, it’s like a ghetto Furby?”** – Regis L

After such inspirational, enthusiastic feedback, I knew he had to be made.

## **THE NAME**

The name is inspired by the Disney movie Big Hero 6. The protagonist is named Hiro Hamada, a young robotics enthusiast. Hamada ‘Hammy’



Baymax from Big Hero 6

The process of inspiration went far and wide. The first ingredient list included:

- *Sugar*
- *Spice*
- *Everything nice*

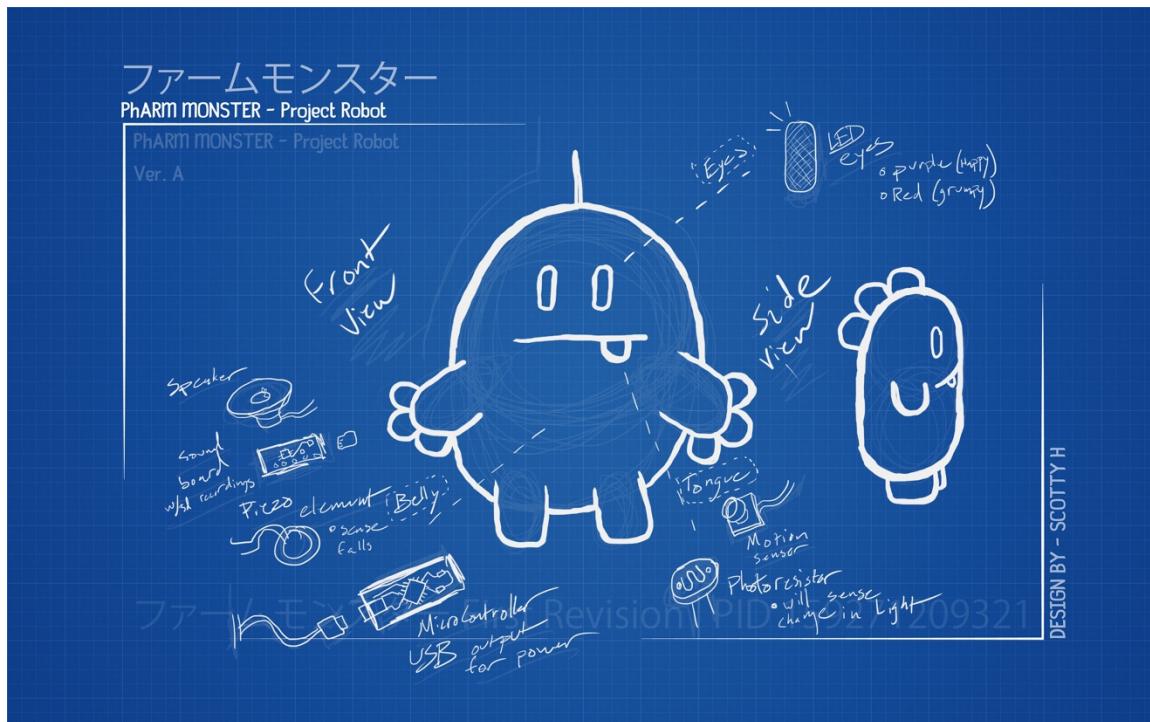
Just as The Professor was creating The PowerPuff Girls, I would need something more...



# Chemical X

The special ingredient would be a cheap OEM portable speaker that screamed 'Made in China' louder than a broken McDonald's Happy Meal toy. Looking past the poor craftsmanship and embracing its dirt-cheap price and appearance, lived a surprisingly well-designed recharging circuit.

## Thus, the PharmacyMonster was born. Kind of.



# INSTRUCTIONS

PharmMonster is rechargeable with a standard phone charger/computer USB port.

## **2X BUTTONS:**

Right Arm = Jokes

Left Arm Quick Press = Quotes (Some are mispronounced by the voice synthesizer, but it's a little comical, so it was not corrected)

Left Arm Hold Press = Oregon Trail game



## **OREGON TRAIL GAME**

Let the directions finish, listen for the choices, the monster will state each choice again separately and after you hear the choice you wish to choose, make a loud sound.

3 Decisions:

Trade choice:

Banker

Carpenter

Farmer (Pharmer) ☺

These change the odds, banker being the highest success rate.

Although the success rate is higher, the score multiplier is lowest for the banker.

Ford the River:

Ford

Float the wagon

Bear Trap:

Chop their arm off

Attempt to free their arm

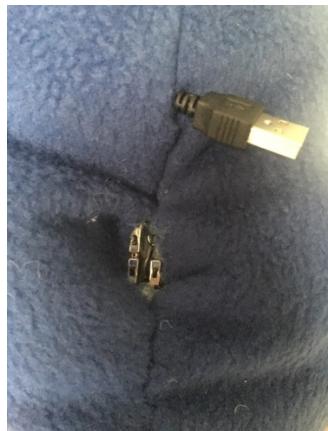
Leave them for dead

After this choice, your score will be calculated. Each choice you made has different percent chances of success based at random. The higher the risk, the great the point reward.

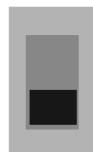
Can you become a 'True Trail champion'?

## 3X SWITCHES

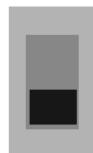
located on back along with USB charging cable



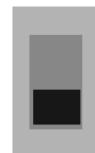
Nothing :)



Disable  
Mic



Power



## 8 BALL MODE

To use 8 ball mode, hold hammy like Simba with his head to the sky and tail to the ground.

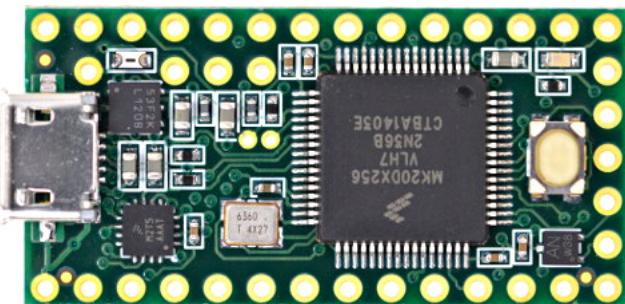


Once it notifies you that 8 ball mode is active, the light will shine. Give him a slight up and down shake to get your answer. (All answers are from the original 8 ball).

# THE PLATFORM HARDWARE

## PROCESSING

The section of the PharmMonster responsible for running the code is a base, modified board from PJRC utilizing a 32-bit ARM processor.



Unmodified Teensy 3.2 from PJRC

## AUDIO

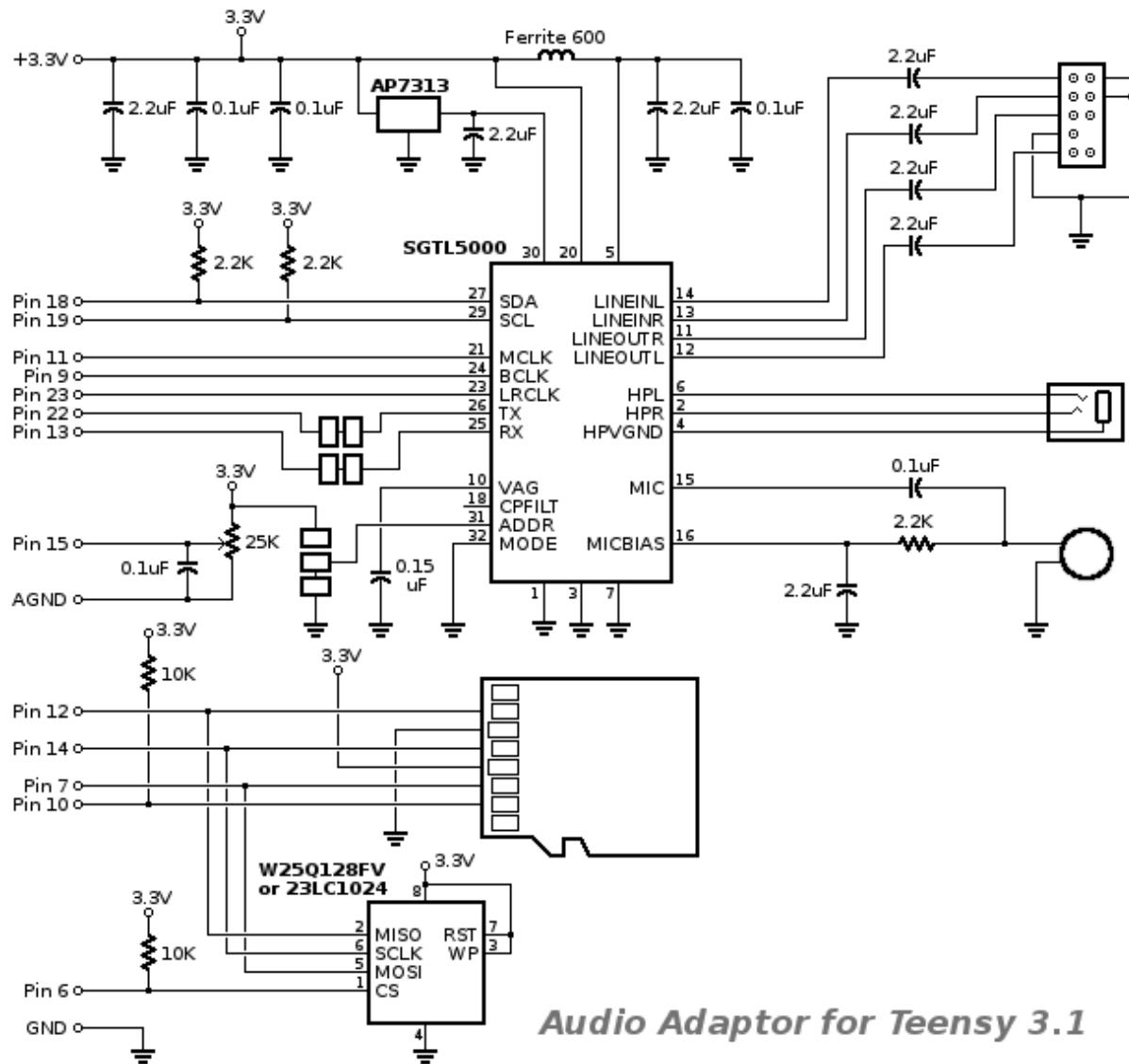
The audio processing is done by a Freescale SGTL5000.

Analog Inputs

- Stereo LINEIN - Support for external analog input

- Stereo LINEIN - Codec bypass for low power
- MIC bias provided
- Programmable MIC gain
- ADC - 85 dB SNR (-60 dB input) and -73 dB THD+N (VDDA = 1.8 V)
- Analog Outputs
- HP Output - Capless design
- HP Output - 62.5 mW max, 1.02 kHz sine into  $16\ \Omega$  load at 3.3 V •
- HP Output - 100 dB SNR (-60 dB input) and -80 dB THD+N (VDDA = 1.8 V,  $16\ \Omega$  load, DAC to headphone)
- LINEOUT - 100 dB SNR (-60 dB input) and -85 dB THD+N (VDDIO = 3.3 V) Digital I/O
- I2S port to allow routing to Application Processor Integrated Digital Processing
- Freescale surround, Freescale bass, tone control/ parametric equalizer/graphic equalizer clocking/control
- PLL allows input of an 8.0 MHz to 27 MHz system clock - standard audio clocks are derived from PLL Power Supplies
- Designed to operate from 1.62 to 3.6 volts

The circuit diagram is provided by PJRC.



**Audio Adaptor for Teensy 3.1**

The 'Pin' labels are all IO interfaces for the Audio processing board.

Although titled for 3.1, it is the same for 3.2 – 3.6.

The support of daisy-chaining the boards is possible, but not implemented in the building of the PharmMonster due to not needing quad-channel output.

All IO are used in the diagram with the exception of Pin 15 to GND which can accept a potentiometer intended to control the volume of the audio output.

# THE PINOUTS

## IO WIRING

### PINOUT WIRING

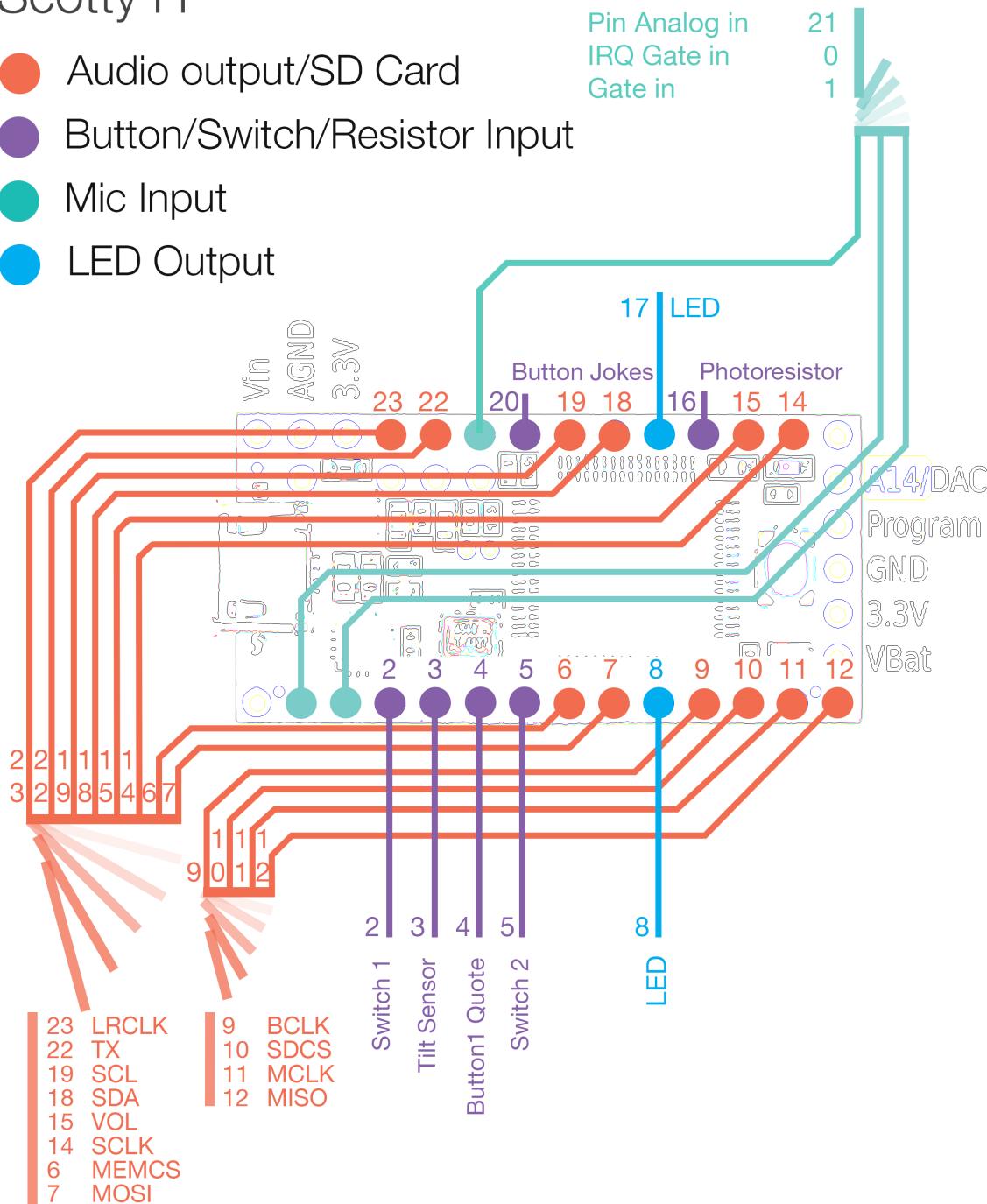
The diagram on the following page shows the wiring from the main board, to the audio board along with basic input listed in the key. The number corresponds to the PJRC documentation for the Teensy 3.2. The following number scheme directly corresponds with the code for the PharmMonster.

All arrays and counts for the PharmMonster are zero-indexed.

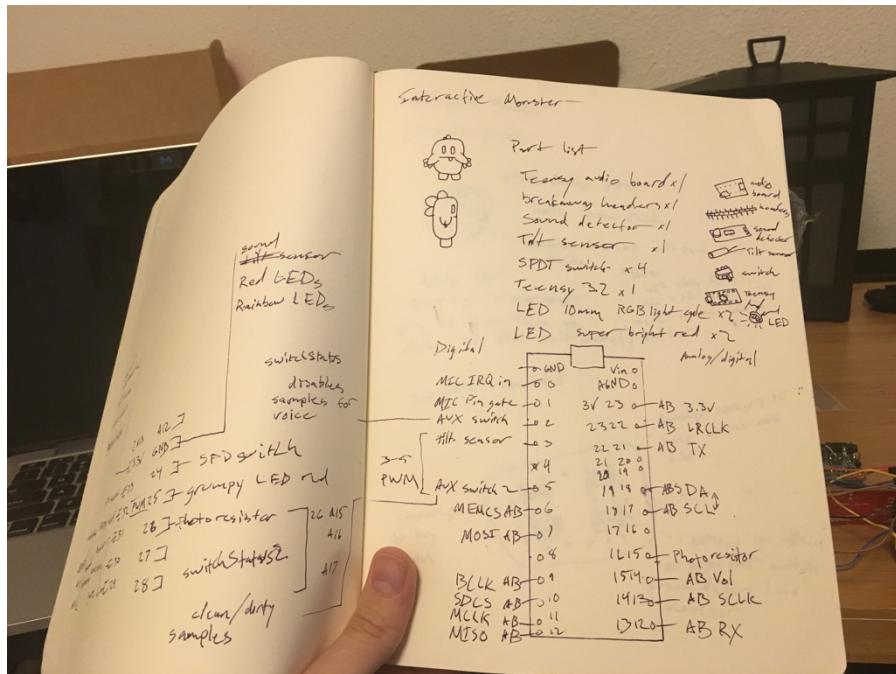
# Robot Monster Teensy 3.2 Pinout Map

## Scotty H

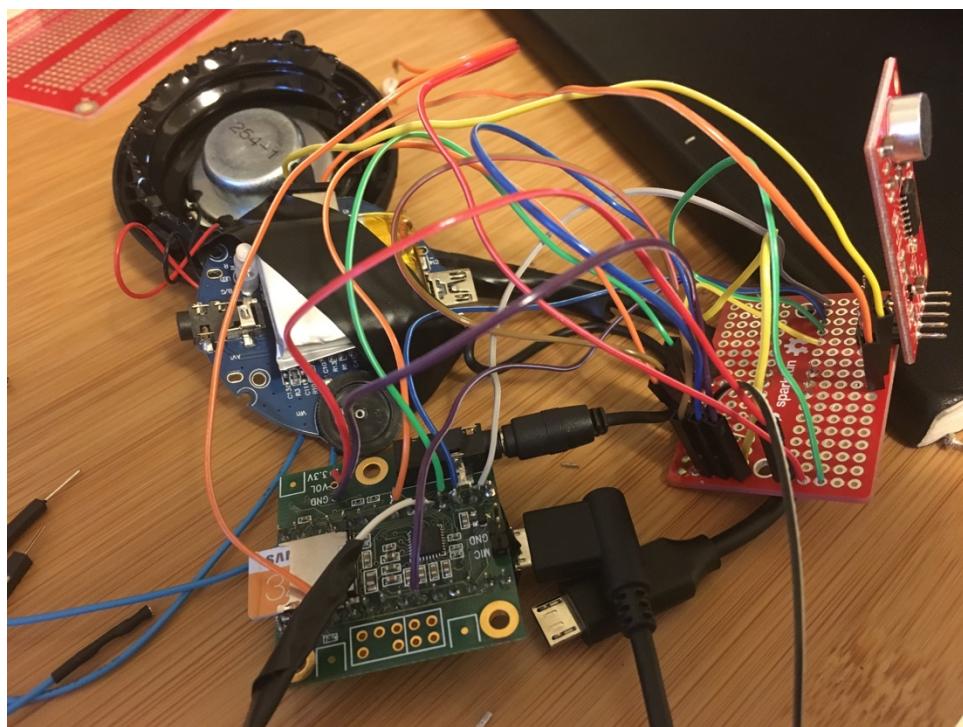
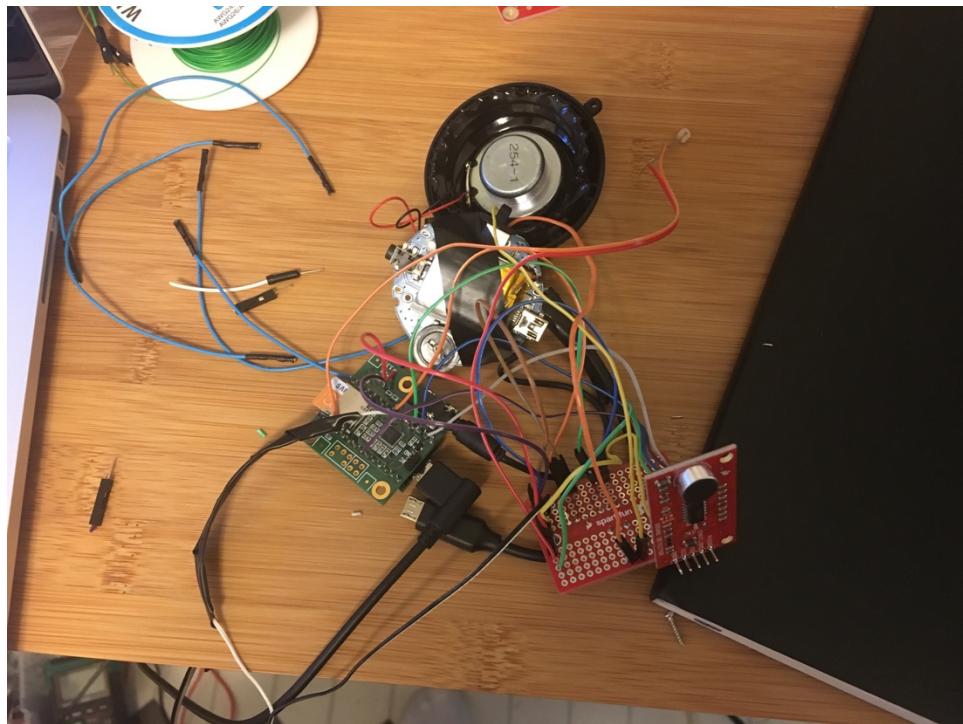
- Audio output/SD Card
- Button/Switch/Resistor Input
- Mic Input
- LED Output



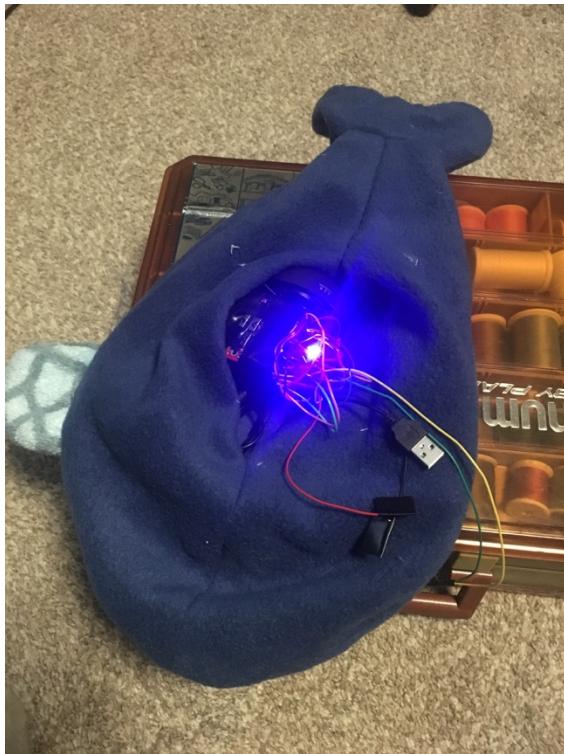
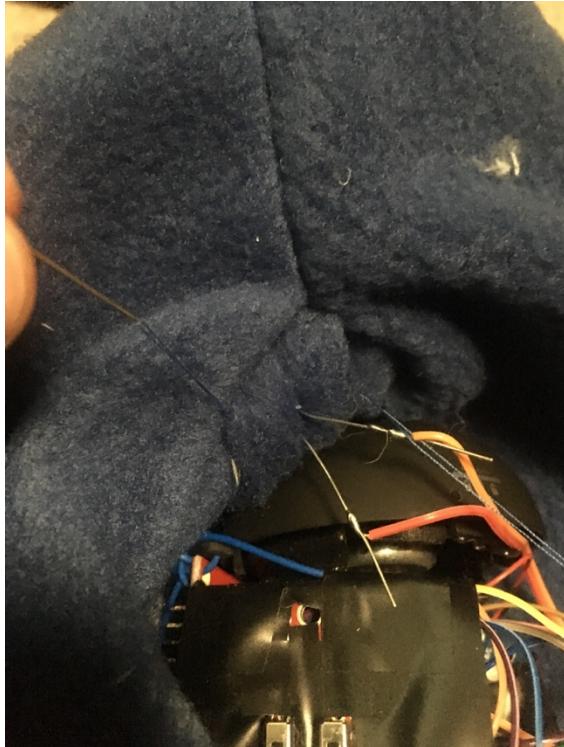
# PROTOTYPING



# WIRING

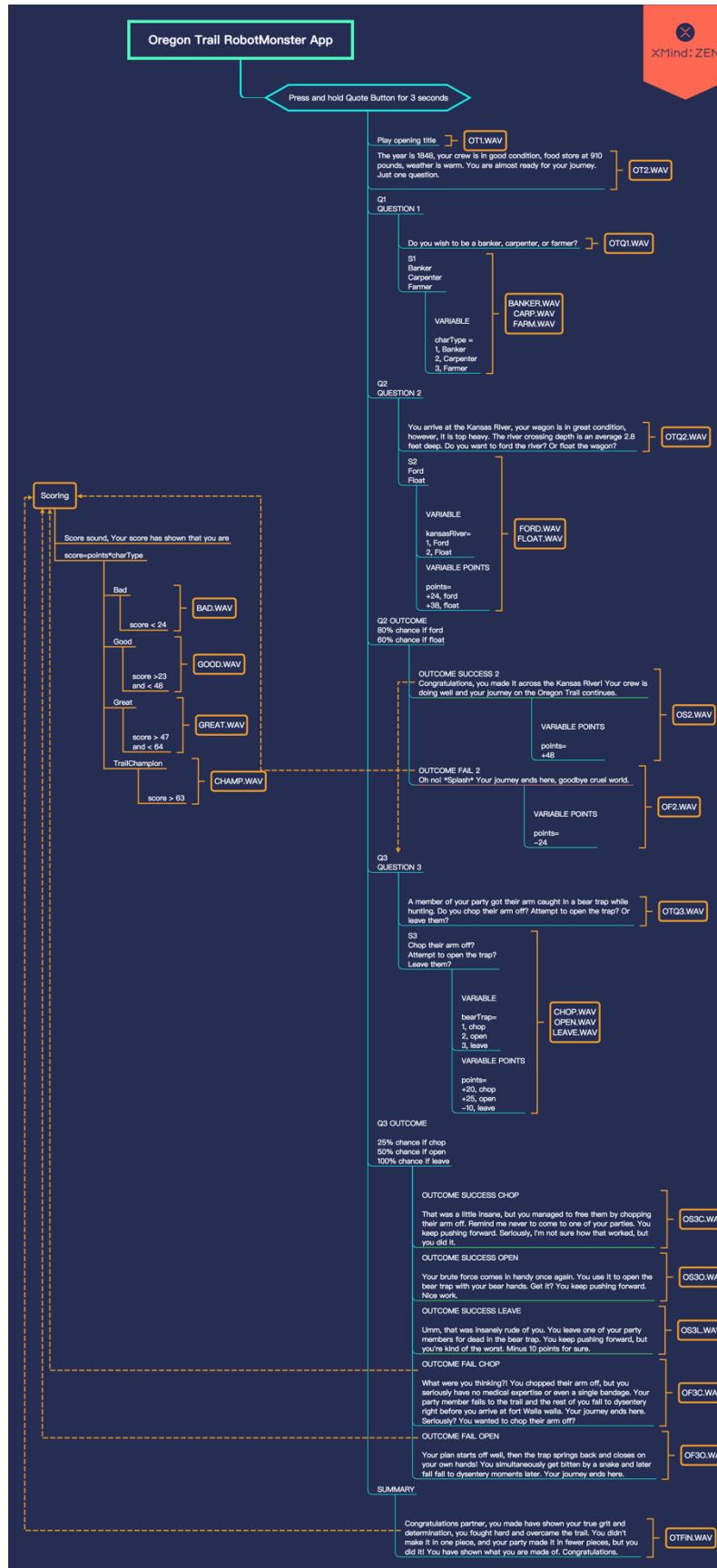


# SEWING



# **THE EASTER EGG PLANNING PROCESS**

Building the Easter Eggs is a standardized process of planning out the pseudocode with XMind. Color coding the audio file names in orange outlines help highlight and organize the process of implementing the actual code.



# AUDIO FILES EXPORTING

## THE PROCESS

Recording takes place in Ableton using a Sunflower virtual audio interface for Mac OS X. This enables the ability to virtually record audio playing on the computer.

## MIDI SETTINGS IN MAC OS X

Optionally, you can select the ‘plus’ icon located in the bottom right of the MIDI audio devices window. You can find these option by searching ‘MIDI’ with a ‘Spotlight Search’.

This will create a ‘Multi-Output Devices. Set the Sample Rate of 44.1kHz, check the ‘Built-in Output’ and ‘Sunflower (2ch)’ device you created. This will allow you to hear what you are recording while not interfering with Ableton. In theory, if you are using windows, you would could not use ASIO audio drivers and it would allow for similar usability with higher latency.

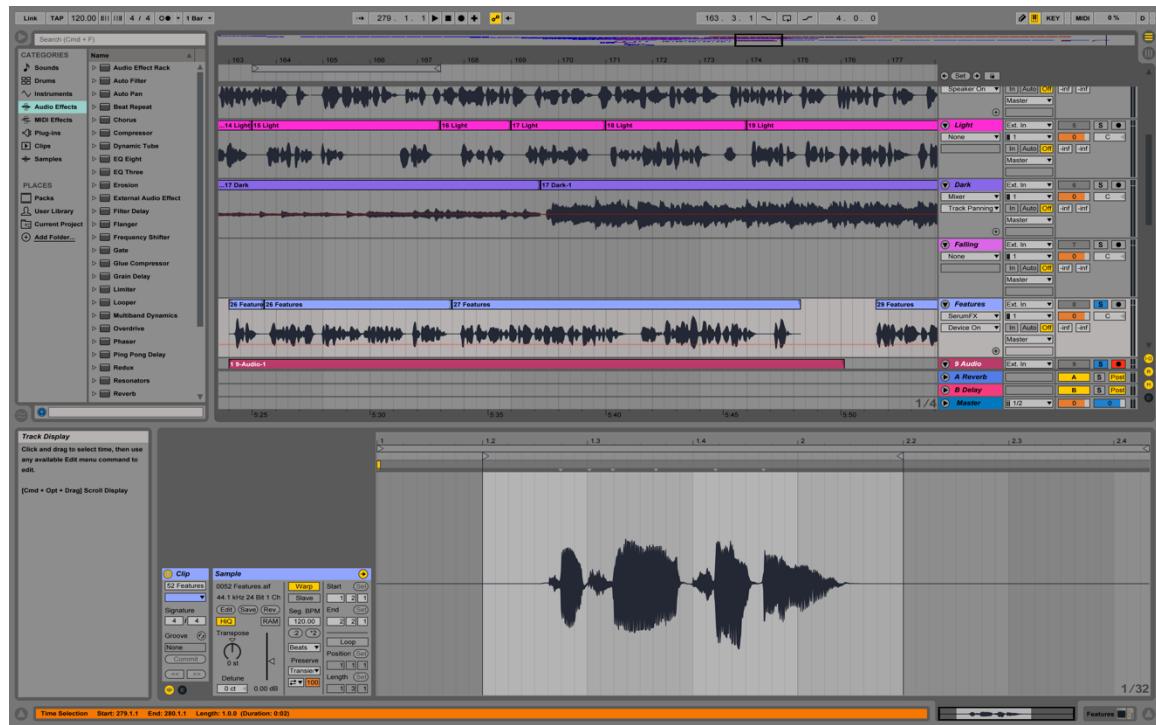


## ABLETON EXAMPLE

General workflow example. You can name each clip with 'Command + r' and choose to export each clip as an uncompressed WAV at 44.1kHz.

### He will not accept compressed audio files

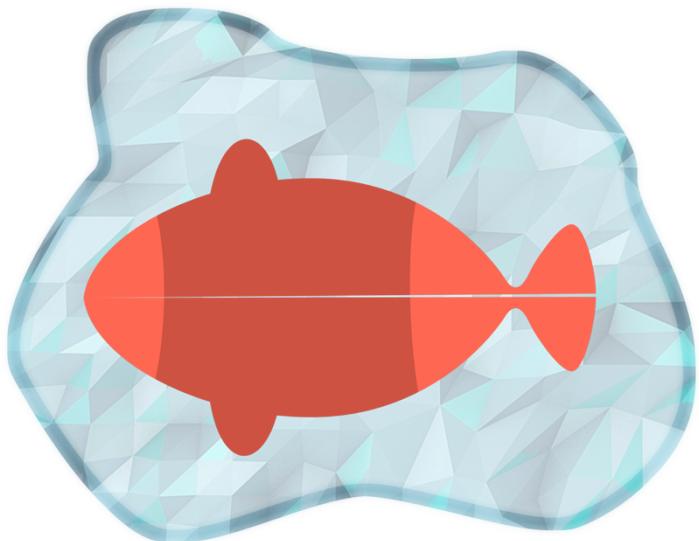
The audio codec, in its current form, can only process lossless WAV files at 44.1kHz. Nothing else.



# DIAGRAM FUNCTIONS

## THE PHARMMONSTER ANATOMY:

- *Tilt sensor*
- *2x Cherry MX Blue keyboard switches*
- *3x JST Switches*
- *Speaker*
- *Photo-resistor*
- *Microphone*
- *Multicolor LED*



# **AUDIO INPUT ACCOUNTING AMBIENT NOISE**

The PharmMonster responds to dB and relies on this rather than content of the audio input such as voice commands. This is to ensure that there is zero risk for voice capture and data leaking. It does not save any data outside of temporary memory that wipes after the device is turned off. The non-volatile memory only stores the WAV files, no user data.

Adjusting for room noise is important to ensure audio triggering from an abnormally loud sound. Each time at startup, the PharmMonster takes ~1000 audio dB levels from the mic and averages the dB level. This takes ~1.01 seconds for V 1.02.

## THE LEVEL

The dB increase needed to trigger the abnormal sound function is 1 magnitude over the baseline.

To ensure that the device can keep up with a room that varies heavily with ambient noise, a new 999 readings are taken after the abnormal noise function is triggered.

To ensure that the speaker does not interfere with the dB level, the readings are taken after the sound clip is done playing. The way the code is structured ensures that it is not possible to take mic levels during the playback of audio.

