

JawDuino (Mike's Talking Skull Project)

This project uses an Arduino and KA2284 LED sound meter module to create a cheap, yet effective talking skull. The cost of this project will vary depending on which type of skull you use, etc.. but the prototype I built while creating this ended up costing about \$20.



KA2284 Power Level Indicator Battery Indicator Pro Audio Indicating Module 5Y

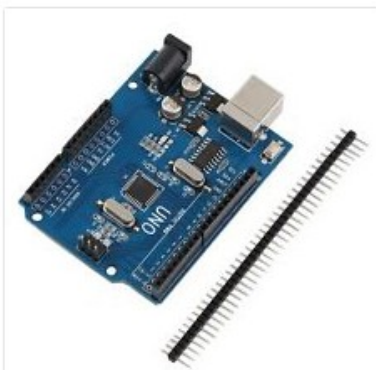
\$0.99

Buy It Now

Free shipping

162 sold

From Hong Kong



UNO R3 ATmega328P Development Board With Boot Loader For Arduino HR

\$3.00

Buy It Now

Free shipping

353 sold

From Hong Kong

You may use any 5v RC servo, but one like this will work for jaw movement.



New MG90S Gear Metal Servo Micro Servo For Boat Car Plane Helicopter

\$2.96

Buy It Now

Free shipping

233 sold

From Hong Kong



5Pcs 30cm Remote Control Servo Extension Cord Cable Wire AD

\$1.60

Buy It Now

Free shipping

From Hong Kong

These servo extension cables are handy and provide a disconnect for your servo



5pcs 3.5mm Female 5 Pins Stereo Headset Interior PCB Mount Audio Jack Socket

\$0.76

Buy It Now

Free shipping

From China

The audio jacks are optional, if you want to run using line-level audio without cutting up an old audio cable (**OR** you could cut up an audio cable, **OR** you could run from speaker-level signals)

The Skull I used is just a cheap \$3.99 Walgreens skull. The price may have gone up \$1 in the last year since I bought mine. I opened the top with an Exacto knife and used electrical tape as a guide. This skull has a nice hinged jaw that is stable and strong.



The linkage you select to connect your servo to the jaw may end up being the most expensive component, depending on what you decide to use. You could borrow parts from an old toy car, etc.. but if you buy real RC airplane linkage, you know the linkage will be strong and trouble free.

I used Dubro Ball Links and 4-40 threaded rod for mine:



Dubro 884 Heavy Duty 4-40 Ball Link

\$2.07

Trending at \$2.53

Buy It Now

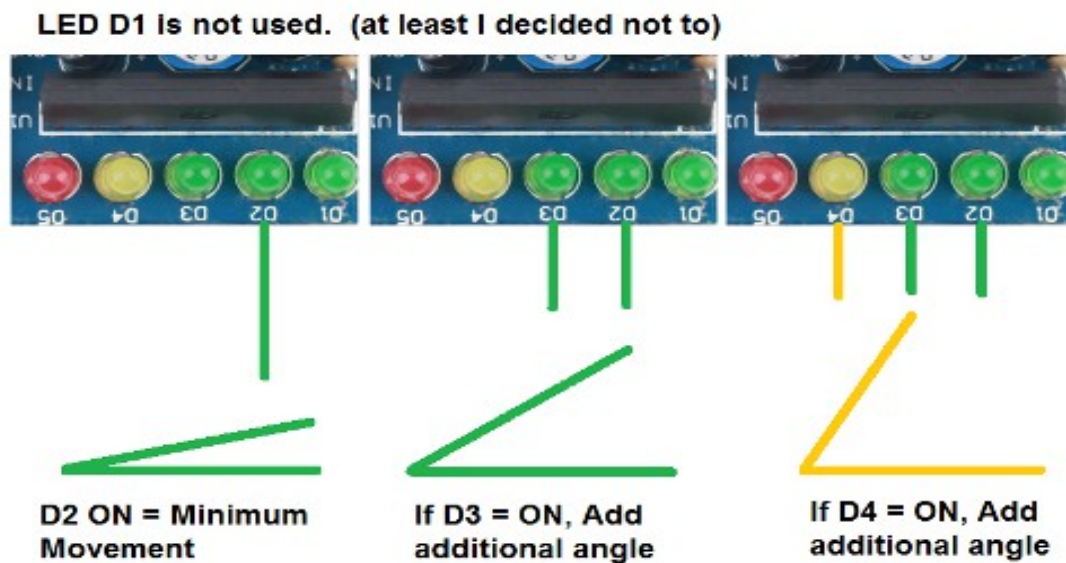
Theory of Operation

The KA2284 LED sound meter module acts like a little 5 segment peak meter that lights up proportionally to the volume of the audio. It takes audio input from one channel.

I have found that line-level audio works well, as does speaker-level audio. There are advantages and disadvantages to each. Using a speaker-level signal means that you would always need to play at the same volume. If you use line-level, you may vary the volume at the amplifier.

Either way, you will want to adjust trimpot on the KA2284 so that the LED's bounce while the audio is playing, and so that the peak/red led only lights up on the loudest of sounds.

We will pick off the led voltages on 3 of the 5 LED's to give a 3 bit resolution servo position:



You could certainly tap all 5 LED's if you wanted to experiment, but I had good luck just using the middle 3 (D2, D3, D4). I decided that skipping D1 would filter out small ambient noise.

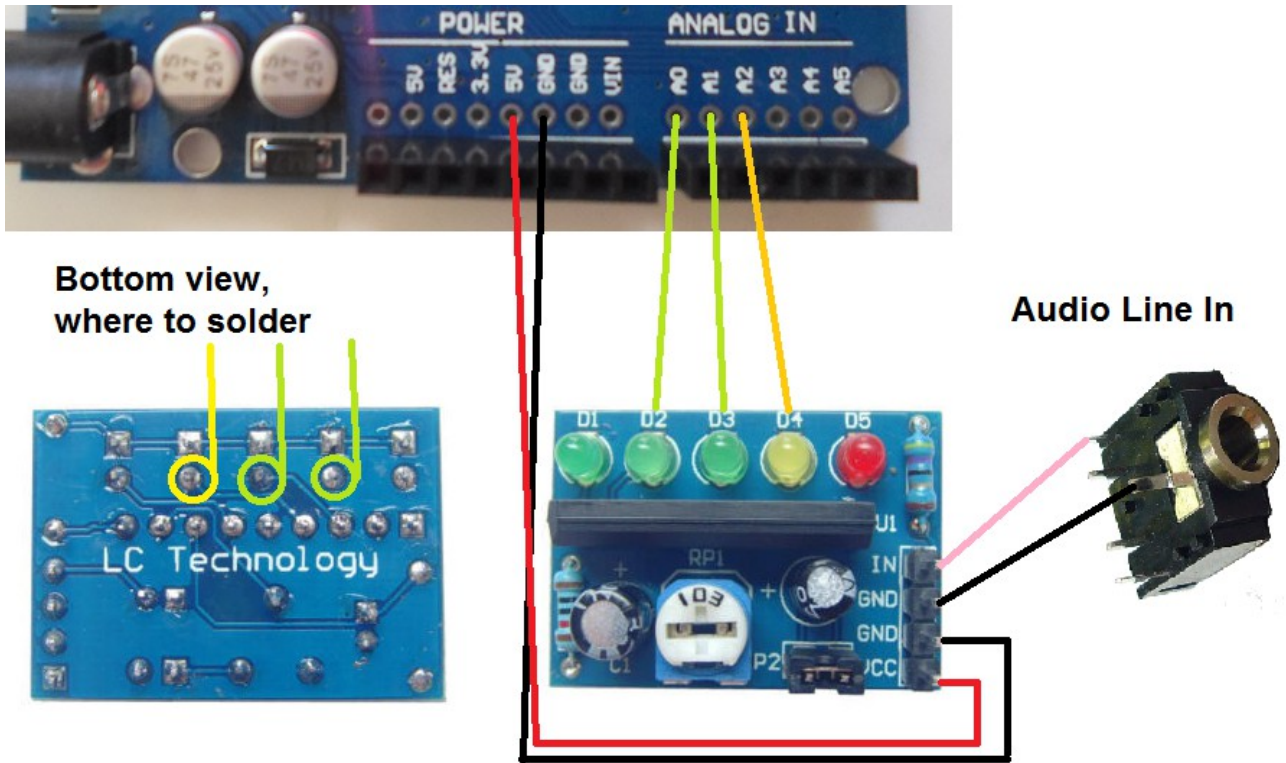
Here is a sample of code where the angle of movement is being increased more and more as each successive LED is found to be on (the analog input voltage goes under a specific threshold)

```
void audio_updates()
{
  audio_value = 0;
  if(analogRead(A0) < 341) audio_value += 60;
  if(analogRead(A1) < 341) audio_value += 60;
  if(analogRead(A2) < 341) audio_value += 60;
}
```

You can see that if all LED's are detected, the sum would be 180. That is the maximum swing (in degrees) that the servo library uses.

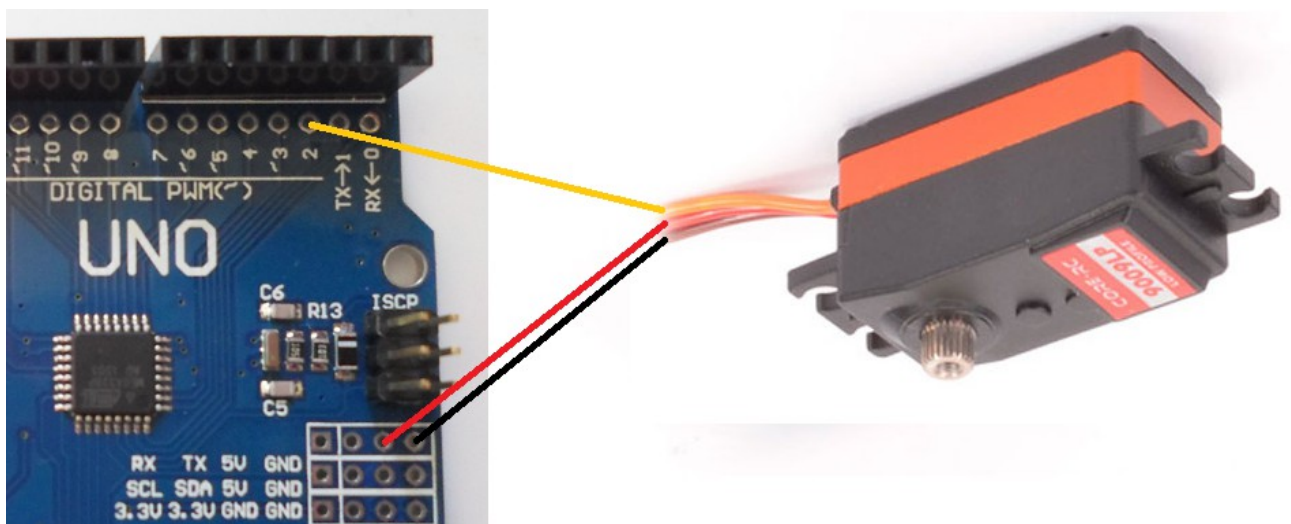
Wiring

Here is how I wired up my prototype:



The upside-down view of the LED meter board is so you can see where I soldered onto the cathodes of D2 thru D4. I picked one of the audio channels at random (right/left) from the stereo audio jack.

***Note- you can also hook this audio meter to a speaker-level audio signal- experiment!



Powering Everything

There are probably a dozen or more ways to power this project, but the simplest will be with a 5vdc **REGULATED** power supply. That is because:

- The Arduino may be run from 5vdc
- The RC servo will require 5vdc
- The KA2284 runs from 5vdc
- Many mp3/audio modules run from 5vdc
- This will take the load off the Arduino's 5v voltage regulator, so you could use a Nano or Pro Mini to save cost and space.

You could also run a 7-12vdc adapter to the Arduino's barrel jack (if you are using an UNO).

Since applications will vary, so will your power requirements.

Before Mounting the Jaw Linkage

It is very important to note that before connecting the jaw linkage, you need to "mock up" the servo mounting in your skull. You may also need to perform a few adjustments to the sketch code in order to adjust the full swing of the jaw servo's range of motion. What this means is that you may not want your jaw servo to travel its full 180 degrees of travel- actually, you probably only want to use about 20 degrees of travel.

To change/tweak the travel, you need to play with these 2 numbers:

Tweak These

```
83 void set_minmax()
84 {
85     //set the first parameter in the following functions to a number between 0 and 180.
86     //I used 92 and 72 in my tests to give about 20 degrees of motion.
87     //You may swap the large and small numbers to reverse direction.
88     //Just play with them, upload the code, then ground pin 11 to run the sweep test.
89     // Be sure to only play with these numbers while the jaw linkage is disconnected,
90     // otherwise, you risk hitting mechanical limits and damaging your linkage or servo!
91     servo.setMinimumPulse(map(92, 0, 180, 512, 2400));
92     servo.setMaximumPulse(map(72, 0, 180, 512, 2400));
93 }
```

Here is a shot of the jaw linkage



**DUBRO BALL
LINK IN JAW**