

Demon Pumpkin

This is a lab guide for creating your own simple animatronic pumpkin. This project encourages students and makers to innovate upon the base design to add their own personal touches.

APPROXIMATE TIME (EXCLUDING PREPARATION WORK): 1 HOUR

PREREQUISITES:

- Setup / Install latest Arduino software from <http://www.arduino.cc/en/Main/Software>. These instructions were tested / verified with version Arduino version 1.6.5
- Download sketch code from <http://github.com/SkoonStudios/DemonPumpkin>
- Although not specifically required, basic soldering skills are useful to make this a permanent project

PART LIST:

Most of these parts can easily be obtained on the Internet through searching, however they are all readily available on sites such as Adafruit or SparkFun*.

- Arduino Uno Rev 3 or similar board based on the ATmega328P chipset.
 - If looking for a smaller form factor board, some of the various wearable boards will work as well (Trinket Pro, etc...)
- Small Pumpkin
 - The larger the pumpkin, the larger the servo to lift the head will need to be
- 5 mm Red LEDs (x 2)
- 220 Ohm Resistor (x 2)
- High Torque Micro Servo (e.g. Tower Pro MG90S)
 - When selecting a servo, be mindful of the operating voltage and the amount of torque that is used to ensure that it is compatible with the Arduino board.
- Battery / USB Power Source

The following parts may be optional depending on your configuration

- Breadboard
- Jumpers / Wiring

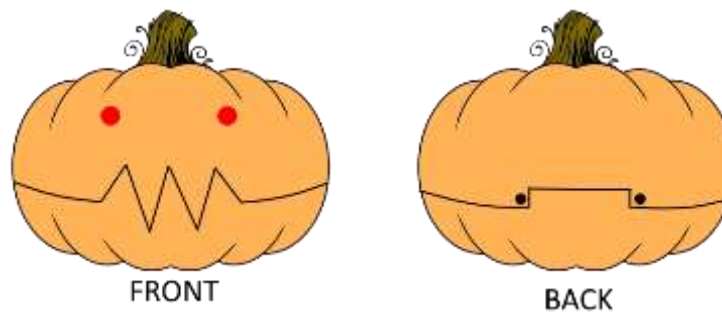
TOOLS:

- Pumpkin Carving Tools
- Drill / Drill Bits
- Wire Cutter / Stripper
- Soldering Iron (if soldering)
- Shrink Wrap / Heat Gun (optional)

INSTRUCTIONS:

A) Carving The Pumpkin

- 1) The pumpkin needs to have the top half completely separate from the bottom half so that the jaw is able to move.



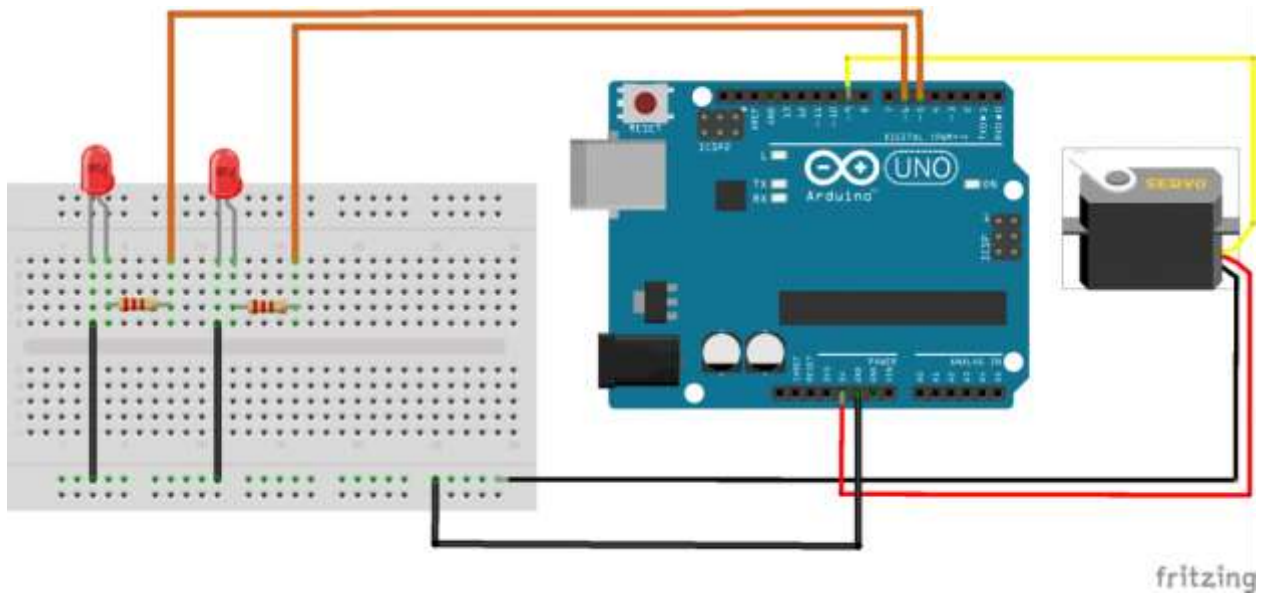
- 2) Scoop out the seeds and goop from the inside of the pumpkin
- 3) Drill holes the size of the LED in the upper half of the front of the pumpkin for the LEDs to rest in. The holes should NOT go all the way through the pumpkin, but rather approximately half way so that the eyes will appear partially recessed in the rind of the pumpkin
- 4) Cut off two 1.5" – 2" sections of the metal coat hanger
- 5) Drill a hole smaller than the diameter of the coat hanger on the back side of the pumpkin (represented by the black dots) going horizontally from the top half into the bottom half. Push the coat hanger through this hole to act as a hinge which will hold the two halves together.



- 6) Mount the servo on the side of the pumpkin bottom with the arm positioned so it can lift the top half.
- 7) Drill a larger hole on the bottom of the pumpkin (opposite of the stem) to feed in hidden wires into the pumpkin.

B) Wiring The Circuit

1) Wire the circuit per the following diagram



2) Some notes about the circuit

- The black wires generally represent connection to ground, the red wires represent a connection to power (+5V), and the yellow / orange wires are control wires although they have power running through them as well.
- Resistors - There are a ton of other websites dedicated to this, but we need to have a resistor here to limit the current across the LED to a safe value. Without a resistor, the LED will quickly overload and pop
- For the LED, the cathode (-) is typically the shorter leg, the anode (+) is typically the longer leg. If you hook up the polarity backwards, the LED will not light up
- The servo can be a big power drain / spike on the circuit. If the battery power is too low, it can cause the Arduino to reset. Additionally, there are some schools of thought that recommend isolating circuits to prevent damage to the Arduino caused by spikes. While I generally agree that is a good idea, in the interest of simplicity here, it has been omitted.
- You will notice that the control pins for the LEDs and the Servo have a ~ besides them. This indicates that they are a Pulse Width Modulation (PWM) channel. More information can be found here if you are interested in reading more.

- 3) The LED legs will stick through the rind of the pumpkin to be wired to the ground and resistors respectively.
- 4) The servo motor can be mounted in the pumpkin

C) Programming The Circuit

- 1) Hook the Arduino board up to the computer via USB cable
- 2) In the Arduino UI, select the appropriate board / COM port
- 3) Load the code sketch downloaded in the Prerequisites section
- 4) Upload it to the board

The Arduino sketch has 2 main functions that are called when the board is first powered on. Initially, the `setup()` function is first called to initialize various components and variables within the program. The setup code here does the following:

- Sets the initial state of the variables (the eyeBrightness, and mouthPosistion). This function also sets up the `_NextEventTime` variable which is compared against the onboard timer to determine when the pumpkin will either come alive next or go back to sleep.
- Initializes the random number generator
- Configures the pins used to control both the LEDs as well as the Mouth Servo

The other function that will then get called indefinitely is the `loop()` function. When it is time for the state to change (either wake up or go to sleep), the following logic occurs.

- The Pumpkin Is Asleep
 - The LEDs will Fade On (`eyeFadeInOut()`)
 - After the LEDs are at full brightness, the timer is set for 20 seconds (defined by `EVENT_INTERVAL_MSEC`) to turn off the effects.
 - The code then uses the random number generator to determine the effects that get used (if the mouth moves, and how the eyes blink).
- The Pumpkin Is Awake
 - Each time `loop()` is called, the minor updates to the eyes and mouth are made by `updateEyeState()` and `updateMouthState()` respectively.
 - At the end of the wake interval, the LEDs will Fade Out (`eyeFadeInOut()`)
 - After the LEDs are completely out, the timer is set for 2 minutes (defined by `WAKE_INTERVAL_MSEC`) to wake up again.

SUMMARY / ACKNOWLEDGEMENTS:

This project was originally created while se to display at the 2015 Charlotte Mini Makers Faire at Discovery Place. It was conceived through various ideas gathered online as a quick, cheap, easy project to demonstrate creative ways to utilize a programmable microcontroller. We encourage you to innovate and put your own creative touches on this project with some ideas may including (but not limited to motion sensor, sounds, and other code innovations for different effects).

Please keep this document fully intact, and if you do build, innovate, or have questions on this project, I do ask that you please let us know at info@skoonstudios.com.



<http://www.skoonstudios.com>

Last Revised: October 18, 2015

* Skoon Studios, LLC has no affiliation or specific endorsement with Adafruit Industries or SparkFun but in the past have bought components and recommended the detailed tutorials and instructions available on the respective sites.