Build a Voice-Controlled Lamp

https://www.sciencebuddies.org/science-fair-projects/project-ideas/Elec_p093/electricity-electronics/arduino-voice-controlled-lamp (https://www.sciencebuddies.org/science-fair-projects/project-ideas/Elec_p093/electricity-electronics/arduino-voice-controlled-lamp)

Procedure PDF Date: 2023-10-26

Experimental Procedure

Note: This engineering project is best described by the **engineering design process,** as opposed to the **scientific method.** You might want to ask your teacher whether it's acceptable to follow the engineering design process for your project before you begin. You can learn more about the engineering design process in the Science Buddies Engineering Design Process Guide (http://www.sciencebuddies.org/science-fair-projects/engineering-design-process/engineering-design-process-steps).

Planning Your Voice Control System

Before you buy parts for your project, you will need to decide what appliance(s) you want to control, and how you want to control it. For example:

- Will you control more than one appliance? What commands do you want to use? For example, if you are controlling a single lamp, the words "on" and "off" are probably sufficient. If you want to control multiple appliances independently, you may need different commands, such as "turn on the light" and "turn on the fan."
- Do you want your voice control system to respond to everyone (speaker-independent) or only to you (speaker-dependent)?
- Where will your appliance, Arduino, and microphone be located? For example, do you want to be able to walk into a room and use your voice to turn on a floor lamp that is across the room? Will you control a desk lamp while sitting at the desk?
- What is your budget? Some of the modules are more expensive than others.
- Is there potential for background noise?

You can always modify your plan later, but make sure you know what you want to do before you start buying parts.

Choosing a Speech Recognition Module

Based on your plan for your voice control system, you will need to choose a speech recognition module for your Arduino. There are many third-party modules available with different features. You can read about the features of each module on its website. (See the Materials section for examples). Before buying one, consider the following:

- Some modules come unassembled and require soldering to attach header pins. If you do not have access to a soldering iron, make sure you purchase a module that does not require soldering.
- Some modules come with built-in commands that you do not need to train.
- Some modules allow you to train new commands and some do not.
- Some modules are speaker-dependent and some allow speaker-independent commands.
- Different modules can store different numbers of commands of varying length. For example, some may allow you to record full sentences, while others may only allow commands that are a second or two long.
- Some modules have an on-board microphone, while other modules have a connector for an external microphone.

Determine what features you need for your voice control system based on your plan. For example, if you want the system to work for everyone in your family, a speaker-independent module may be better. (If you use a speaker-dependent module, you will need to train each command separately for each person.) Some of the microphones that are built into or come with the speech recognition modules have a fairly short range. They may only work well when you are very close to them. That could be fine if you will use the device while sitting at a desk. If you prefer to be able to control an appliance from across the room or from different directions, a module that lets you connect a higher-quality or omnidirectional external microphone may be better.

There is no single "right answer" for which speech recognition module to buy. They all have different features and different prices. Which one is "best" for your project will depend on your criteria and constraints. Choosing one is part of the engineering design process.

The following two sections contain instructions for using the Elechouse Voice Recognition Module V3, which allows you to train multiple speaker-dependent commands, and the Grove Speech Recognizer, which comes pre-loaded with 22 speaker-independent commands but does not let you train new commands. If you choose another speech recognition module, you will need to follow the instructions for the module you purchase.

Using the Elechouse Voice Recognition Module V3

Watch this video for an overview of how to set up and use the Elechouse Module V3, or follow the written steps below.

1. Solder header pins to the GND, VCC, RXD, and TXD pins.
2. Plug the microphone into the microphone jack.
3. Plug the header pins into a breadboard. Alternatively, you can use male-female (M-F) jumper wires to connect the module directly to the Arduino without using a breadboard.
4. Use jumper wires to make the following connections to the Arduino (Figure 2).
   a. Elechouse GND to Arduino GND
   b. Elechouse VCC to Arduino 5V
   c. Elechouse RXD to Arduino digital pin 3
   d. Elechouse TXD to Arduino digital pin 2
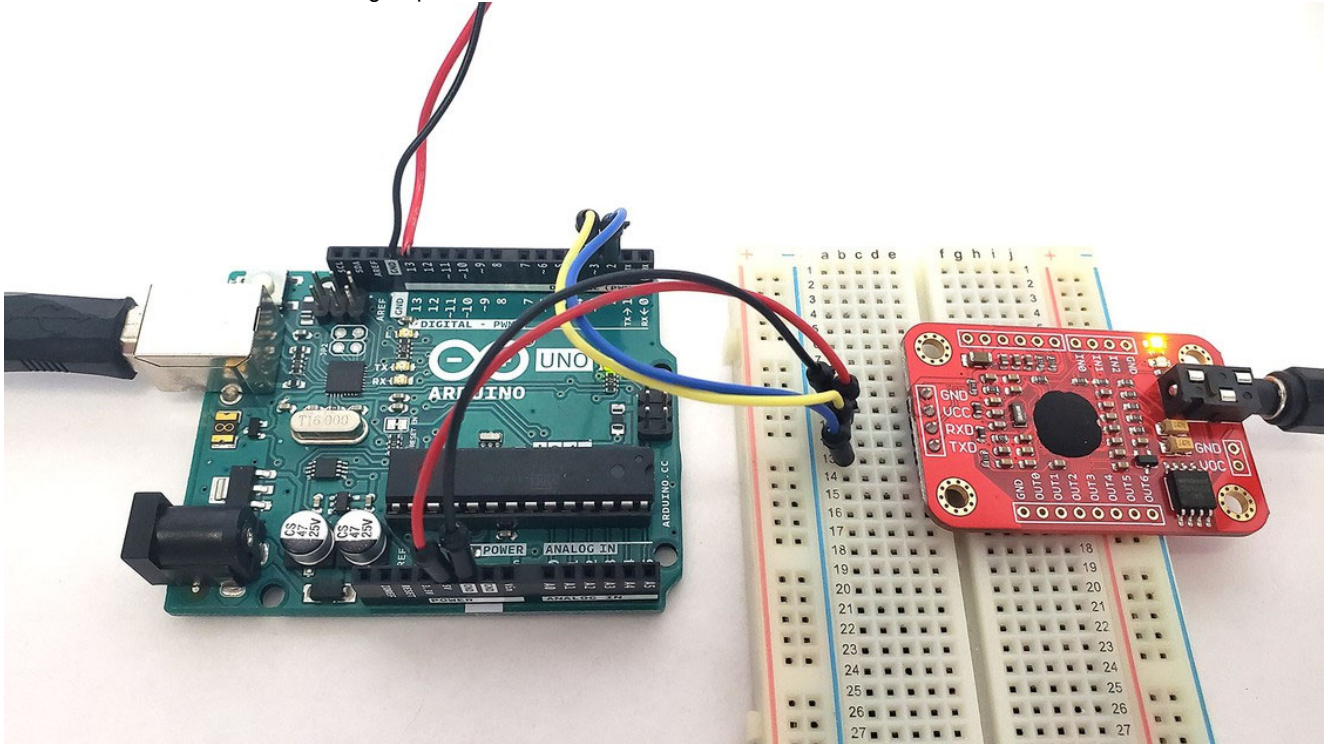
**Figure 2.** Elechouse Voice Recognition Module V3 connected to an Arduino UNO.

5. Download the Arduino library from the Elechouse website (https://www.elechouse.com/product/speak-recognition-voice-recognition-module-v3/).
6. Extract the .zip folder to the Arduino libraries folder on your computer. This folder may be saved in a location such as C:\Program Files (x86)\Arduino\libraries.
7. Connect your Arduino to your computer and open the Arduino Integrated Development Environment (IDE).
8. Select *File > Examples > VoiceRecognitionV3 > vr_sample_train*.
9. Under *Tools*, make sure you have the correct Arduino board and COM port selected.
10. Upload the program to your Arduino.
11. Select *Tools > Serial monitor*.
12. Set the baud rate to 115200.
13. Type "train 0" in the box at the top of the serial monitor and press Enter.
14. Watch the serial monitor. When it prints "speak now," say "On" clearly.
15. When it prints "speak again," say "On" again.
16. If the two recordings do not match, the program will continue prompting you to speak. Keep repeating the phrase until it prints "success."
17. Enter "train 1" then repeat steps 14–16 for the word "Off."
18. Enter the command "load 0 1". This loads the two commands you just recorded into memory for use with voice control.
19. Try saying "On" or "Off." The serial monitor should print out "VR Index," followed by the index number of the matching command (0 for "On" and 1 for "Off"). If it does not print anything, make sure you are speaking clearly and at the same distance from the microphone as when you recorded the command.
20. Select *File > Examples > VoiceRecognitionV3 > vr_sample_control_led*. Upload the program to your Arduino.
21. Try speaking the words "On" and "Off" to control the Arduino's onboard LED (the small LED labeled "L" near Arduino pin 13).
22. You can now use your voice recognition module to control a single appliance by connecting pin 13 to the IoT relay (#Iot-relay).
23. To control multiple appliances, you will need to train additional command words following the procedure above and modify the vr_sample_control_led program to control additional Arduino pins.

Using the Grove Speech Recognizer

Watch this video for an overview of how to set up and use the Grove Speech Recognizer, or follow the written steps below.

1. Plug the included cable into the 4-pin header on the Grove Speech Recognizer board.
2. Use jumper wires to connect the wires in the cable to your Arduino (Figure 3):
    a. Black wire to Arduino GND
    b. Red wire to Arduino 5V
    c. Yellow wire to Arduino pin 2
    d. White wire to Arduino pin 3
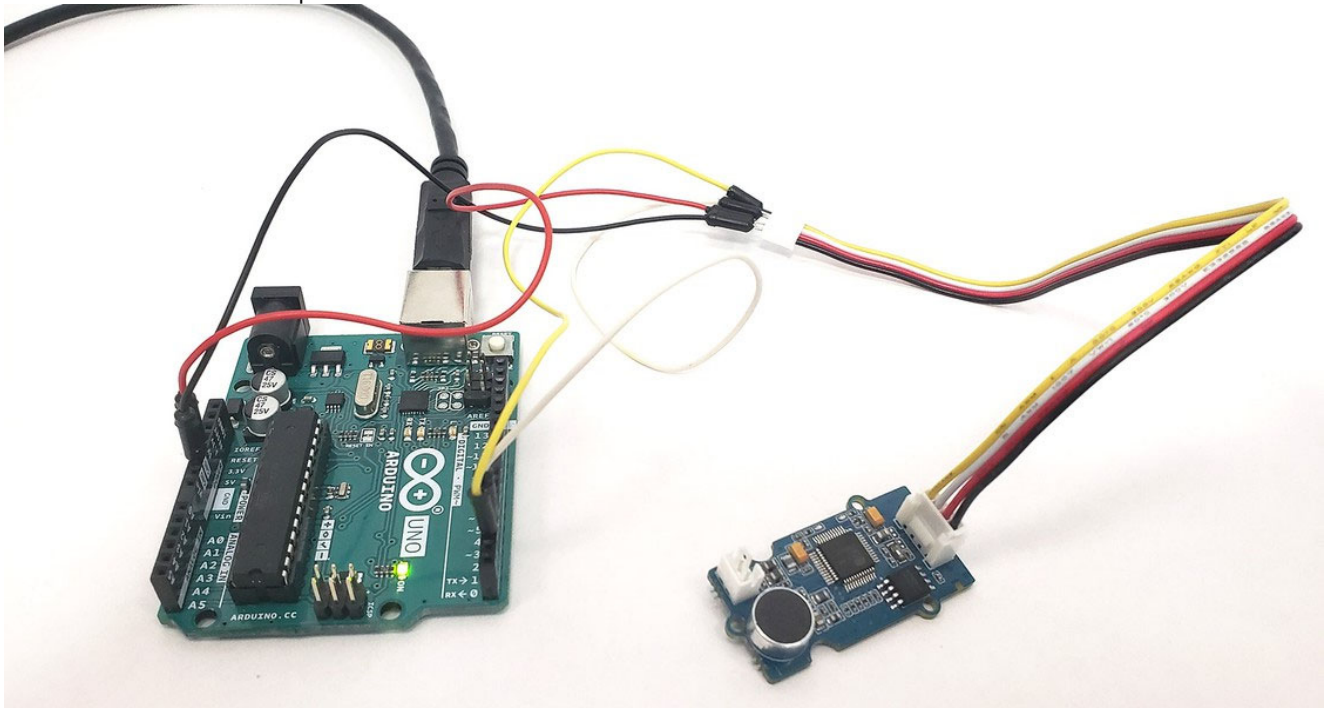
Figure 3. Grove Speech Recognizer board connected to the Arduino.

3. Go to the Seeed Studio wiki page (https://wiki.seeedstudio.com/Grove-Speech_Recognizer/), scroll down, copy the example code, and paste it into your Arduino IDE.
4. Under *Tools*, make sure you have the correct Arduino board and COM port selected.
5. Upload the program to your Arduino.
6. Select *Tools > Serial monitor*.
7. Set the baud rate to 9600.
8. Say the wake word "hicell" (the words "hi" and "cell" pronounced together as one word). The red LED on the board should light up when it recognizes the wake word, indicating that the board is waiting for a command.
9. Once the red LED is lit up, speak one of the command words or phrases. There are 22 commands listed in the example program — for example, "Start" and "Stop." If the board recognizes the command, the blue LED should light up and it will print the command to the serial monitor.
10. Download grove_control_LED.ino (http://www.sciencebuddies.org/cdn/Files/18845/6/grove_control_LED.ino) and upload it to your Arduino. This is a slightly modified version of the example code that uses the commands "Turn on the light" and "Turn off the light" to control the Arduino's onboard LED.
11. Try saying the wake word followed by "Turn on the light." The Arduino's onboard LED (the one labeled "L" near pin 13) should turn on. Then say the wake word followed by "Turn off the light" and the LED should turn off. Remember that you need to say the wake word again for each new command. If it does not work, make sure you are close to the microphone and speaking clearly, then try again.
12. You can now use the speech recognizer to control a single appliance by connecting pin 13 to the IoT relay (#Iot-relay).
13. To control multiple appliances, you will need to further modify the code as shown in grove_control_LED.ino to make use of additional commands and other Arduino pins.
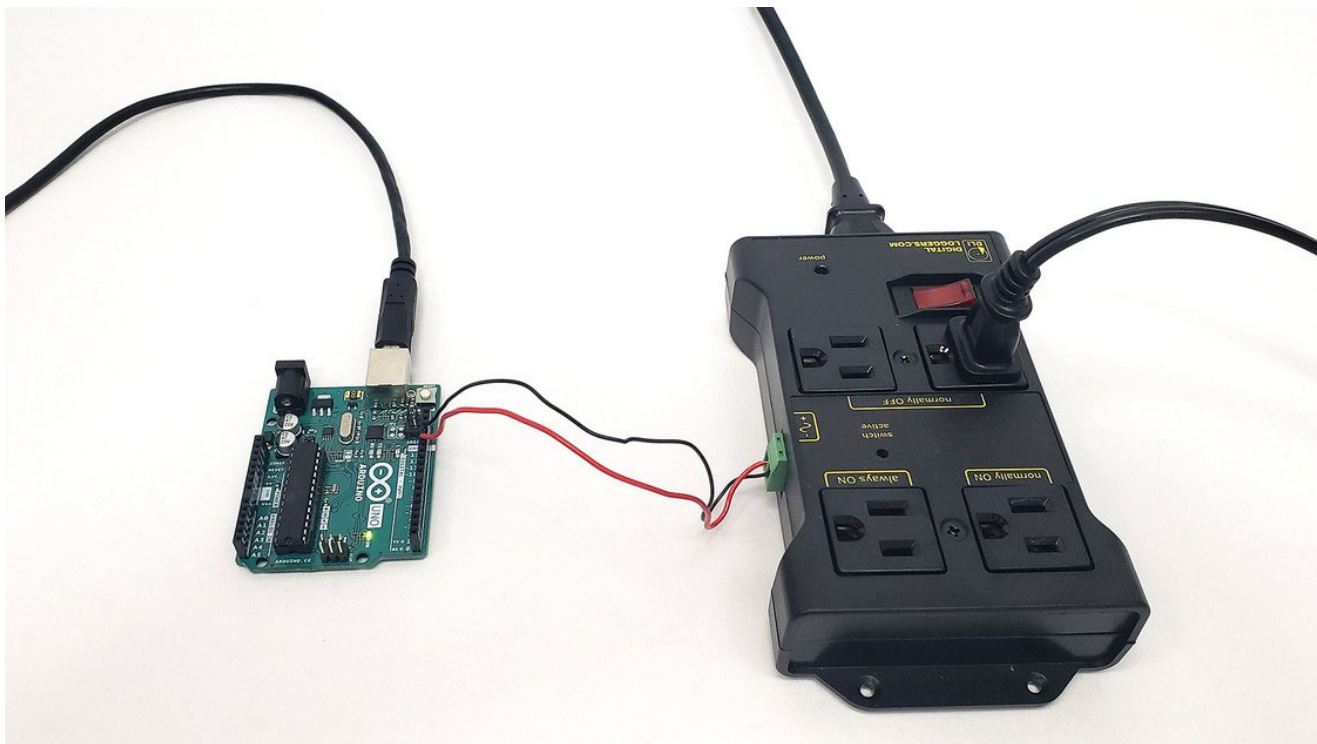
Using the IoT Relay

Caution: The IoT relay allows you to safely control a high-voltage appliance that plugs into a wall outlet using a low-voltage control signal from a microcontroller like the Arduino. Remember that you should never cut or modify the high-voltage wires (like the power cord for an appliance) directly.

Watch this video or follow the written instructions below to learn how to set up and use the IoT relay:

https://www.youtube.com/watch?v=p2yaFh5kuvw (https://www.youtube.com/watch?v=p2yaFh5kuvw)

1. Pull the green screw terminal connector out from the side of the IoT relay.
2. Loosen the screws, insert one end of a jumper wire into each terminal, then re-tighten the screws.
3. Connect the "+" wire to Arduino pin 13 and the "-" wire to GND. (See Figure 4.)
4. Make sure the power switch on the IoT relay is in the Off position.
5. Plug an appliance, like a lamp, into one of the "normally off" outlets on the IoT relay.
6. Upload the Blink example code to your Arduino (*File > Examples > 01.Basics > Blink*).

7. Turn on the power switch on the IoT relay. Your appliance should turn on and off repeatedly.
8. Turn off the power switch on the IoT relay.

Figure 4. IoT relay connected to the Arduino with jumper wires.

Build Your Smart Home Device

Using the information in the sections above, you should now be ready to set up your own voice-controlled smart home appliance.

1. Set up your voice recognition module. You can follow the instructions in one of the sections above, or follow the instructions that came with your module if you purchased a different one.
2. Test your voice commands and use them to control the Arduino's onboard LED so you can confirm that they are working.
3. Connect your Arduino to the IoT relay and plug in an external appliance so you can switch it on and off.
4. Depending on what you want your project to do, test it in different scenarios. For example, do the voice commands work for everyone in your household? Do they work from across the room or if there is background noise?
5. Look at your test results and decide if your solution meets all your design requirements. If not, what can you do to improve it? Do you need to re-train any of the commands or move the microphone to improve performance? Do you need to revisit the requirements to get the results you envisioned?
6. Keep track of all your test results and make notes on any changes so you can communicate your work to others.