Arduino Controlled/Voice TV Remote

by **jordantallent** on June 18, 2013

Table of Contents

Arduino Controlled/Voice TV Remote 1
Intro: Arduino Controlled/Voice TV Remote 2
Step 1: Materials 2
Step 2: Setup/IR code retrieval 3
File Downloads 3
Step 3: Arduino Library for Your Remote 3
File Downloads 4
Step 4: Arduino Emulation Code 4
File Downloads 4
Step 5: PC Desktop Application 5
File Downloads 5
Step 6: Expansion Into Android 5
Related Instructables 6
Advertisements 6

Author:jordantallent

I am a student of Western Carolina University, about to be a senior (it is the summer of 2013). I currently work as a sales intern for a local company that deals mostly in hydraulics and heavy industry distribution. While at school I work as a lab assistant. Really all I do is support for the engineering department, on the electrical side. Some classes require students to design and fabricate a circuit, and I would be the one to actually build it on the quick circuit router that we have at school.

Intro: Arduino Controlled/Voice TV Remote

Being a college student, you can become pretty lazy. This was the motivation behind my project for making a new remote for my TV that could be controlled from my phone, computer, or my voice.

I started this project my freshman year in college, before I even really knew what an Arduino was, and now I see it as a super simple project that anybody with an Arduino and a computer can do.

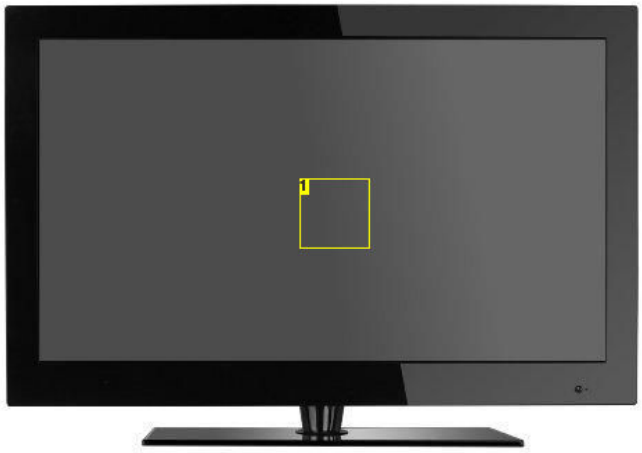


Image Notes
1. TV that I want to control

Step 1: Materials

-Regular IR remote control that works on your device
-IR receiver
-IR LED 940nm
-any Arduino
-some jumper wires
-Computer
-Visual Studio(optional)
-Bluetooth SPP app for Android(optional)

Image Notes
1. Remote control that I want to emulate

Step 2: Setup/IR code retrieval

First off I would like to point you in the direction of where I got all of this information from, here . Thanks to Adafruit and Ladyada for all of their great tutorials! I would suggest reading that tutorial to get a more in depth idea of what is involved in this project.

After reviewing the supplied code in the tutorial, I modified it slightly so that when the serial terminal printed what it read from the IR sensor, I could just copy and paste it into the corresponding function in my emulator code. I have all of my source code attached in this Instructable.

The first step in finding out what signals your remote is outputting, we need to connect your Arduino to an IR sensor as shown in the picture.

After this is done, we can upload some code to Arduino and have it output the on/off cycle for each button on your remote. Upload the 'IR_Read' from my source. When this is running, open up a serial terminal, point your remote at the IR sensor and press one of the buttons that you will want to emulate. The output should look something like this...

.....(put output picture here)

We need to get this output for every button that you plan to emulate. I would open a word document or notepad and and copy the serial output and post it somewhere that you can access it later, and be sure to label it so you know which button it corresponds to.

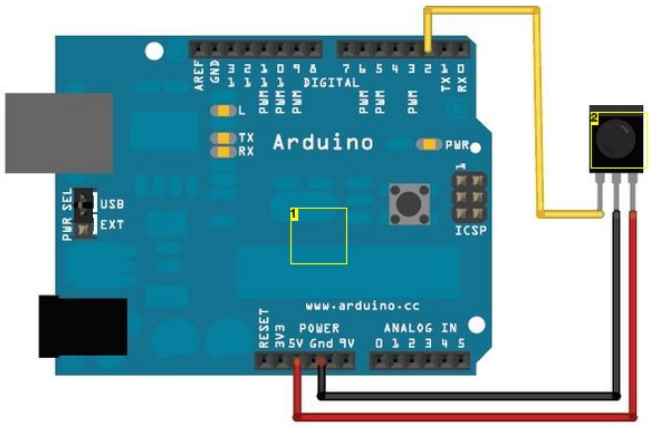


Image Notes
1. Arduino or compatible
2. IR sensor

File Downloads

Arduino Sketch.zip (19 KB)
[NOTE: When saving, if you see .tmp as the file ext, rename it to 'Arduino Sketch.zip']

Step 3: Arduino Library for Your Remote

To emulate the codes, I decided to create my own Arduino library instead of putting all of the code in the actual sketch. After getting all of the on/off codes from the previous step, this part is simple. The library consists of two main functions that turn the IR LED either on or off for whatever the correct length is. There are also multiple other functions that are simply a list of the on/off times for each remote control button. If you use my supplied code, all you have to do is replace the guts of each of my functions, such as 'power', with the codes that you copied from earlier. Make sure that you save the library files in the correct Folder so that your Arduino IDE can see it.

File Downloads

Arduino Library.zip (38 KB)
[NOTE: When saving, if you see .tmp as the file ext, rename it to 'Arduino Library.zip']

Step 4: Arduino Emulation Code

This is quite possibly the easiest part of the whole project. The sketch simply waits for a specified character from a serial input, then calls a certain function from the previously created library. That's all there is to it! I just set each of my remote control buttons up with a keyboard button. The only hardware needed for this circuit is an IR LED and appropriately sized resistor based on the specifications on your LED. If you are hooking up a standard 940nm IR LED you could just connect it to pin 13, as most Arduino units have a built in resistor on that pin. You should now be able to control your TV or other IR controlled device right from your Arduino setup, using any serial terminal application.

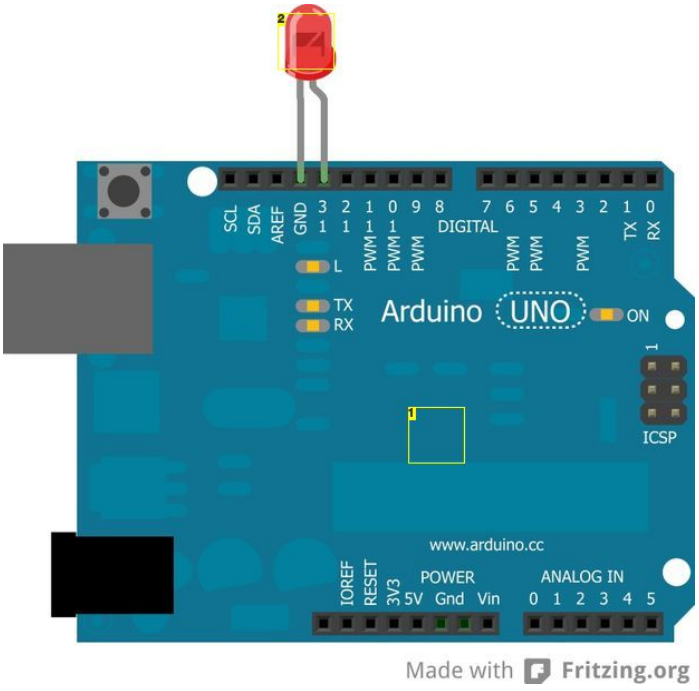


Image Notes
1. Arduino or compatible
2. 940nm IR LED

File Downloads

Arduino Sketch.zip (19 KB)
[NOTE: When saving, if you see .tmp as the file ext, rename it to 'Arduino Sketch.zip']

Step 5: PC Desktop Application

Now that I have the working Arduino remote running using a standard serial terminal, I wanted a friendlier way to use the remote, so I decided to create a small desktop application for my Windows computer using Visual Basic. I wanted a user interface that looked something like a regular TV remote and was easy to use. I made a small program that just had buttons for the numbers, volume buttons, channel buttons, last, mute and power. This is seen in the pictures, also included in a list box so you can choose which COM port your device is on.

After making the plain, boring program that just had buttons, I wanted to get lazier. I have done a little bit of work with Windows voice recognition so I decided to include that into this program as well. This allowed me to just yell at my computer and it would change my TV channel or anything else that I could program in.

File Downloads

Visual Basic Source.zip (4 MB)
[NOTE: When saving, if you see .tmp as the file ext, rename it to 'Visual Basic Source.zip']

Step 6: Expansion Into Android

I have not done too much work in Android programming, but I do know a little bit about using my resources. This led me to find a serial terminal app for Android phones that allows you to connect to a bluetooth enabled Arduino and control the device. This is simply done by attaching a cheap eBay bluetooth module to the RX and TX pins on the Arduino. The app that I use is Bluetooth SPP. Of course, if you are more advanced in Android development, you could make a custom app that looks like a remote, similar to the one that we made in Visual Basic.

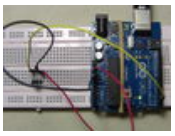
I hope you enjoy! If anyone decides to develop an Android app I would love to see it!!

Related Instructables

iAndroidRemote - Control Android mobile using an Apple Remote by sudar

How to Use IR Remotes with Arduino by Enjoying Electronics

TV Volume Loudness Guard using Arduino by techbitar



Arduino IR remote control decryption (video) by techbitar

Universal IR Remote Power Switch for PC by TimHepner

Arduino-powered A-10 stick grip remote w/Emergency Party Button! by spikec