Automatic Channel Changing TV Remote With Arduino

by ankaki

The initial motivation for this project was that our client has vascular dementia and is legally blind. This has caused her to have difficulties with remembering when and on what channels the television shows that she likes are, as well as seeing the small buttons on a traditional remote. For this reason, our team worked to create a remote, which had a small number of very large buttons, which the user who is legally blind could either see or feel. The remote also needed to be able to automatically change the channel to specific channels at specific times so that even if our client forgot, she would be able to see her shows.

Our solution utilized an Arduino and an infrared emitter to communicate with the television. The Real-time clock module was used to keep track of time so that the remote was able to switch to the desired shows. One large button was also used for the purpose of turning the TV on and off. Also, a buzzer module was attached to alert the user that the channel was being changed.

Competitor Analysis:

We analyzed 3 other adaptive remotes for the criteria required for this project

1. Flipper Remote - a simplified remote with a reduced number of extra-large buttons

Pros: Affordable (only $35) and buttons are larger than on a traditional remote.

Cons: Cannot automatically change channels, and while buttons are larger than a traditional remote, they may still be too small.

2. Logitech Harmony Elite - smart remote which enhances the capabilities of the traditional remote and allows for some customization.

Pros: Supports voice commands and has an intuitive touchscreen interface

Cons: Expensive ($350), cannot change the channels automatically, and has small buttons.

3. Caavo Control Center - smart remote and control center pair which allows connections between the television, the remote and some third party software

Pros: Supports voice commands

Cons: Expensive ($160), cannot change channels automatically, and has small buttons

Supplies:

1. "Arduino" Uno with USB cable - 12.99 from Amazon.com

2. YL-44 Buzzer Module (buzzer module, low-level trigger) - $3.98 w/ shipping from aliexpress.com

3. Real-time clock module (with necessary battery) - $11.50 for 3 from Amazon.com

4. Large Arduino arcade button - $9.95 from adafruit.com

5. Arcade button wires - $4.95 from adafruit.com

5. Infrared emitter and receiver set - $13.99 from Amazon.com

7. 9-volt battery - $10.99 for 8 from Amazon.com

8. 9-volt battery to male DC adapter - $4.99 for 5 from Amazon.com

The external case was 3D Printed using PETG filament

https://www.instructabl...

Download

Step 1: Connecting the Pieces

The ground pin on the piezo buzzer was connected to ground on the Arduino, and the I/O pin was connected to the digital 8 port.

The ground pin on the Real Time Clock was connected to ground on the Arduino, the VCC pin was connected to a voltage pin on the Arduino, the SDA pin was connected to the SDA pin on the Arduino, the SCL pin was connected to the SCL pin on the Arduino.

The ground pin on the IR emitter was connected to ground on the Arduino, the VCC pin was connected a voltage pin on the Arduino, and the DAT pin was connected to the digital 3 port.

The pins on the arcade button were connected to the digital 2 port and the ground pin on the Arduino.

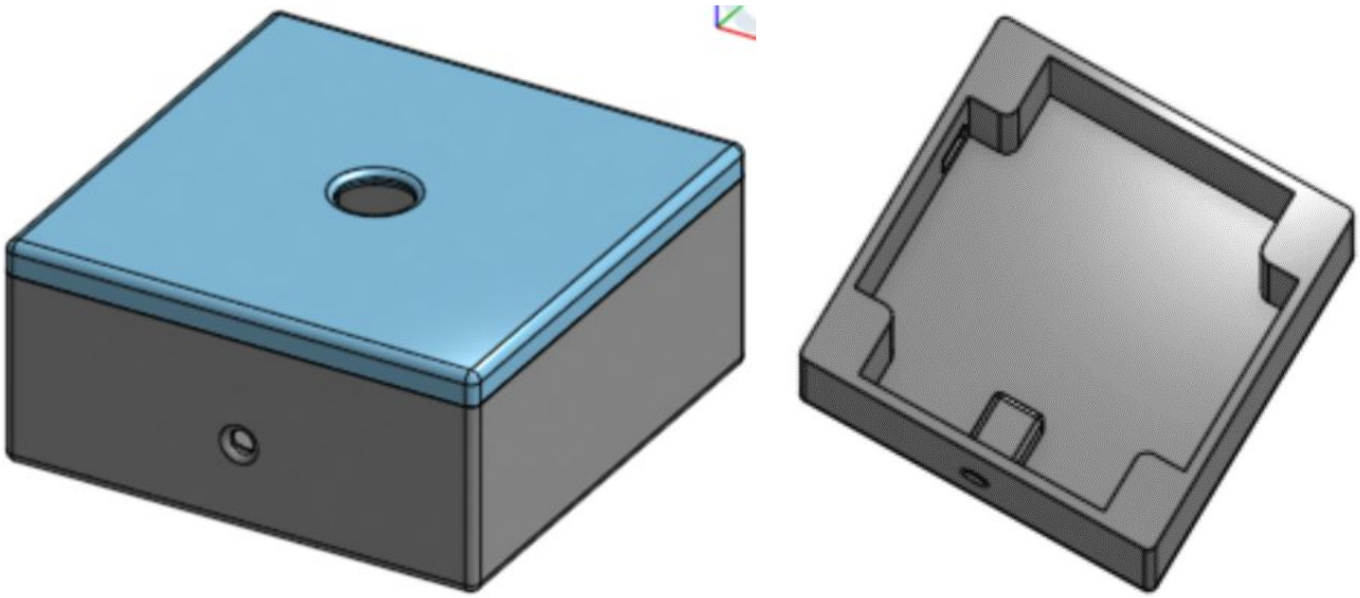
Step 2: The Code

The code for this project can be found here.

Step 3: The Casing

As mentioned before, the casing for this device was printed using PETG filament. Pictured above are screenshots of the CAD files which show what the finished device will look like. The printable STL files both for the base and the lid

are also attached.



https://www.instructabl... Download

https://www.instructabl... Download

Step 4: Use

Instructions on how to use the device:

In order to set up this remote to work with your specific television, you first need to find out what IR codes transmitted from your current remote to the TV. Steps to do this can be found here. Once these codes are known, the specific IR codes needed to be implemented in the Arduino code which is stored on the GitHub. Comments in the program will lead you to exactly where these codes should be inputted.

Once that is set up, the device is very easy to use; simply press the large button on the top to turn on and off the TV. If the TV is on during one of the times that you have preprogrammed for the channel to change, it will do so automatically. When you are ready to turn off the TV, press the button again.

Safety Measures:

Due to the fact that the device is not watertight and there are many wires, batteries, and other electronic components inside of the casing, it is important that the device is **kept dry.**

In addition, due to the fact that this project utilizes Arduino, it is important that the conditions in which it is operated remain within the safe range of **-40 to 85 degrees Celsius.**

Care and Maintenance:

Once the remote is set up, there is not much in means of maintenance that needs to be done on a regular basis. Occasionally, the main 9-volt battery which powers the Arduino and other components will need to be replaced; to do this, remove the current 9-volt battery from the adapter in the casing, and attach a new battery to the adapter. The Real Time Clock module has its own external battery (3V), so even if the main battery dies, it should continue to keep the time. If this secondary battery is to die, however, it needs to be replaced *and* the RTC needs to be recalibrated. This recalibration can be done by plugging the Arduino into the computer and initializing the code.

Step 5: Improvements and Extension Projects

If you are looking for a challenge, there are many other ways to extend this project to your liking! A few possible projects you could try are:

- making an app so that the user/caretaker can change the selected channels on their own
- connecting the Arduino to the internet to get more detailed data, such as time that is accurate to the millisecond
- utilizing public information available about each television channel to give the user more details

Step 6: Resources and References

Big Button Jumbo Universal TV Remote for Seniors. Retrieved from https://flipperremote.com

Control Center Smart Remote + Home Theater Hub - Pay As You Go Plan. Retrieved from https://caavo.com/products/control-center

IR Remote Signals. Retrieved from learn.adafruit.com/ir-sensor/using-an-ir-senso

Logitech Harmony Elite Advanced Universal Remote, Hub and App. Retrieved from https://www.logitech.com/en-us/product/harmony-el...

Samuel123abc. (2017, October 08). Arduino YL-44 Buzzer Module. Retrieved May 22, 2020, from https://www.instructables.com/id/Arduino-YL-44-Buz...

Universal Remote 101: How Do Universal Remotes Work? Retrieved from caavo.com/blogs/news/universal-remote-101-how-do-universal-remotes-work

z3t0. (n.d.). z3t0/Arduino-IRremote. Retrieved from https://github.com/z3t0/Arduino-IRremote/wiki/Rec...