Using an IR Remote with a Raspberry Pi Media Center

Created by Simon Monk

https://learn.adafruit.com/using-an-ir-remote-with-a-raspberry-pi-media-center

Last updated on 2021-11-15 05:58:24 PM EST
# Table of Contents

<table>
<thead>
<tr>
<th>Sections</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview</td>
<td>3</td>
</tr>
<tr>
<td>Parts</td>
<td>3</td>
</tr>
<tr>
<td>• Part</td>
<td></td>
</tr>
<tr>
<td>• Qty</td>
<td></td>
</tr>
<tr>
<td>Hardware</td>
<td>5</td>
</tr>
<tr>
<td>LIRC</td>
<td>6</td>
</tr>
<tr>
<td>Configure and Test</td>
<td>8</td>
</tr>
<tr>
<td>Using Other Remotes</td>
<td>10</td>
</tr>
</tbody>
</table>
Overview

In this tutorial, you will learn how to use an Infrared remote with a Raspberry Pi configured as a media center.

The IR receiver is attached to the GPIO connector on the Raspberry Pi.

Before tackling this project, you need to follow this tutorial (https://adafru.it/c2S) to set up your Raspberry Pi as a media center.

Parts

To build this project, you will need everything from the Media Center setup tutorial (https://adafru.it/c2S) and the following items.
<table>
<thead>
<tr>
<th>Part</th>
<th>Qty</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>IR Sensor</td>
<td>1</td>
<td><a href="http://adafruit.com/products/157">http://adafruit.com/products/157</a></td>
</tr>
<tr>
<td>IR Remote</td>
<td>1</td>
<td><a href="http://adafruit.com/products/389">http://adafruit.com/products/389</a></td>
</tr>
</tbody>
</table>
Hardware

The IR sensor has just three pins, that will connect with three pins on the GPIO connector. To do the connecting, we can use female to female jumper leads. These make a good reliable connection as the IR sensor has unusually thick leads for an IC.

Make the connections as shown below. Note that you do not have to use the same colored jumper wires. But selecting adjacent wires that are still in a 'ribbon' will help keep things neat.
Note that the IR sensor chip needs to be operated at 3.3V not 5V when used with the Raspberry Pi.

LIRC

The interface between the hardware and the Raspberry Pi media centre is managed by a piece of software called LIRC (Linux Infrared Remote Control). This is pre-installed on most recent Raspberry Pi distributions and is included in the Rasbmc distribution, so there is nothing to install, however, there is some setting up to do.

To make sure that the IR hardware is correct, we can connect to the Raspberry Pi running Rasbmc using SSH, which is automatically enabled on this distribution.

If you have not connected to a Raspberry Pi using SSH before, please see this tutorial. (https://adafruit.it/cag)

You can find the IP address of the Raspberry Pi using the XBMC System Info page.
To be able to test the IR receiver without XBMC, you need to make sure that the IR remote feature is turned off, or you will not be able to use LIRC from the SSH. So run the Rasbmc Settings program and make sure that the option Enable GPIO TSOP IR Receiver is disabled.

If you needed to change this you will need to reboot.

Now connect to the Raspberry Pi using SSH and issue the commands shown below:
Now hold the remote in front of the receiver and you should see a series of 'pulse' / 'space' messages appear each time you press a button.

Congratulations! The IR receiver is working.

---

**Configure and Test**

Now that we know that the hardware is okay, we need to give LIRC a config file to tell it about the keys on the remote that we are using.

From the SSH session, issue the command:

```bash
nano lircd.conf
```

... and then paste the following text into it, before saving the file by clicking CTRL-x then Y.

```plaintext
# Please make this file available to others
# by sending it to &lt;lirc@bartelmus.de&gt;
# this config file was automatically generated
# using lirc-0.9.0-pre1(default) on Thu Mar 14 14:21:25 2013
# contributed by
#
# brand:                       /home/pi/lircd.conf
# model no. of remote control:
# devices being controlled by this remote:
#
begin remote
name /home/pi/lircd.conf
bits 16
```

© Adafruit Industries
flags SPACE_ENC|CONST_LENGTH
eps  30
aeps  100

header  8945  4421
one  594  1634
zero  594  519
ptrail  598
repeat  8949  2187
pre_data_bits  16
pre_data  0xFD
gap  106959
toggle_bit_mask  0x0

begin codes
    KEY_VOLUMEDOWN  0x00FF
    KEY_PLAYPAUSE  0x807F
    KEY_VOLUMEUP  0x40BF
    KEY_SETUP  0x20DF
    KEY_UP  0xA05F
    KEY_STOP  0x609F
    KEY_LEFT  0x10EF
    KEY_ENTER  0x906F
    KEY_RIGHT  0x50AF
    KEY_KP0  0x30CF
    KEY_DOWN  0xB04F
    KEY_BACK  0x708F
    KEY_KP1  0x08F7
    KEY_KP2  0x8877
    KEY_KP3  0x48B7
    KEY_KP4  0x28D7
    KEY_KP5  0xA857
    KEY_KP6  0x6897
    KEY_KP7  0x18E7
    KEY_KP8  0x9867
    KEY_KP9  0x58A7
end codes

end remote

This file should be saved in the home directory for the user pi.

Now, return to the Rasbmc Settings program and enable the option Enable GPIO TSOP IR Receiver. At the same time, change the GPIO Remote Profile as shown below:
Restart XBMC and when it has rebooted, you should see a small popup message in the bottom right corner like the one below.

You should now find that your IR remote control will work and that you no longer need the keyboard and mouse to control XBMC.

Using Other Remotes

I generated the config file for this remote using a utility that is part of LIRC called 'irrecord'.

If you have a different remote, then you can generate a config file for it using this tool.

The process is as follows:

- Turn the remote off on XBMC using Rasbmc as we did before using 'mode2'.

- Rename the existing lircd.conf out of the way

- Type the command 'irrecord --list-namespace'. This will tell you the allowed key names that you can use when prompted.

- Type the command 'irrecord -d /dev/lirc0 ~lircd.conf'

- Follow the instructions to the letter. It all seems a bit odd, but the program has to work out the timings and encodings used by the remote.
Found const length: 106959
Please keep on pressing buttons like described above.
-----------------------------------------------
Space/pulse encoded remote control found.
Signal length is 67.
Found possible header: 8945 4421
Found trail pulse: 598
Found repeat code: 8949 2187
Signals are space encoded.
Signal length is 32
Now enter the names for the buttons.

Please enter the name for the next button (press <ENTER> to finish recording)
KEY_VOLUMEDOWN

Now hold down button "KEY_VOLUMEDOWN".

Please enter the name for the next button (press <ENTER> to finish recording)
KEY_PLAYPAUSE

Now hold down button "KEY_PLAYPAUSE".

Please enter the name for the next button (press <ENTER> to finish recording)