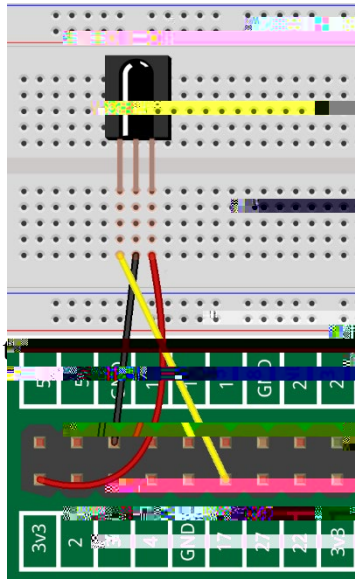
Control-Using-LIRC-for-the-Raspber/

This handout assumes the Raspberry Pi is running Raspbian Buster. Also a working knowledge of the terminal, basic commands from the command prompt and editing text files is assumed.

Wire the infrared received as shown in the diagram. Note that with the raised lens area facing you, the pin on the left is the output and goes to pin 17 of the Raspberry Pi. The pin on the right is power and goes to 3v3. The center pin goes to ground.



1) Open terminal window and install LIRC. Be forewarned that this will likely raise an error "Failed to start Flexible IR remote input/output application support" as the installed files have .dist appended and removed as shown below.

```
sudo apt-get update
sudo apt-get install lirc
```

Don't worry if you got the error message. Just rename the file as shown.

```
sudo mv /etc/lirc/lirc_options.conf.dist /etc/lirc/lirc_options.conf
```

2) Reinstall lirc now that the lirc_options.conf file has been renamed

```
sudo apt-get install lirc
```

This time there will be no error message.

Edit /etc/lirc/lirc_options.conf as follows:

```
sudo nano /etc/lirc/lirc_options.conf
```

This work incorporates material under the Creative Commons BY-SA -

You will need to change two lines.
Locate the line that says “driver” and change “devinput” to “default”
Locate the line that says “device” and change “auto” to “/dev/lirc0”

Press **ctrl-o** and **ctrl-x** to save and exit.

Type the following program into the Thonny program window. Save and run it.

In the program listing above, note the line that says `print(command)`. At that point in the program, the command has been decoded and can be used in an IF statement. When you run the code below, pressing a button on the remote causes the

```
# https://www.instructables.com/id/Easy-Setup-IR-Remote-
Control-Using-LIRC-for-the-Ra/

from lirc import RawConnection
from gpiozero import LED

redLED = LED(18)
redLED.off()
conn = RawConnection()

def ProcessIRRemote():

#get IR command
#keypress format=(hexcode, repeat_num, command_key, remote_id)
    try:
        keypress = conn.readline(.0001)
    except:
        keypress=""
    if (keypress != "" and keypress != None):
        data = keypress.split()
        sequence = data[1]
        command = data[2]
        #ignore command repeats
        if (sequence != "00"):
           return
        print(command)
        if command == "KEY_POWER":
            redLED.toggle()

print("Starting Up...")
while True:
      ProcessIRRemote()
```