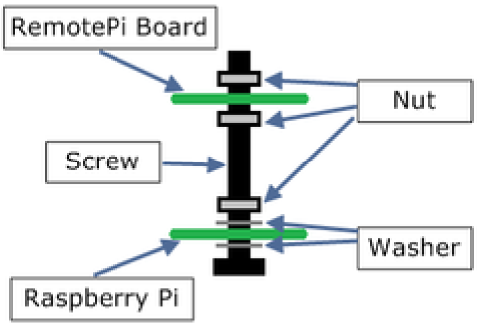
Installation

- Install the shutdown script on your mediacenter installation.
- Switch off and disconnect the power plug from the RPi
- Mount the supplied screw, washers and nut on the RPi's existing mounting hole near the front.
- Plug in the RemotePi board, make sure the GPIO connectors line up
- Adjust the nut under the RemotePi board until the board is levelled
- Secure the RemotePi board with another nut from the top



- Connect power to the RemotePi board's Micro USB connector, do not use the Raspberry Pi power Micro USB connector. If you connect the power by accident to the Raspberry Pi connector, you will not damage anything, but the power control function of the RemotePi will not work.
- You will see a short red/green blink of the RemotePi's LED to indicate the board is working.
- Modify your existing RPi case, so that the micro USB power connector and the button on top of the RemotePi is accessible, and the LED and infrared receiver are visible from outside.
- If you want to use your own existing remote to control your mediacenter as well, then you will have to enable LIRC and

create your own lircd.conf file. (If you use our remote we have already created the file for you and you only need to install it). This file tells the OS how to understand the keys on your remote. You can use the configuration guide on the remote page as a guideline, but you will still need to figure out the lircd.conf contents for your own remote as this file is different for every remote model. There are various tutorials on the web on how to do this, basically you either find a ready-made lircd.conf file or you run a program on the RPi which learns the key functions and creates a lircd.conf file for you.
Please Note : If you create your own lircd.conf, please do not include the KEY_POWER key in the list. The power is handled by the RemotePi Board.

Using the RemotePi Board

- Choose the button on the remote you want to use to switch your RPi on and off in the future, then press the hardware button on top of the RemotePi Board for about 15 seconds until you see the LED blinking green and red. Now you have about 30 seconds time to aim the remote towards the RemotePi Board’s infrared receiver and press the button you want to use for controlling the power of your RPi. You will see the green LED flashing when the button was learned. If no infrared command is learned within 30 seconds you see the red LED flash and the learning mode is exited without changing the current configuration.The learned configuration is remembered (i.e not lost), even when the power is disconnected from the RemotePi Board.
- A short press of the button on top of the RemotePi Board power toggles the RPi in the same way the learned button on the remote

does. While powering up, the LED flashes green and while powering down it flashes red for about a minute, during this time you cannot issue another power toggle request. This is to prevent cutting the power during startup / shutdown.

- Configure the mediacenter system you are using with a script which is shutting it down when the signal from the RemotePi Board is received.
- *After you have installed the script here is how you know that everything works correctly* : When you switch off the power of the RPI using either the hardware button on top of the RemotePi Board or the remote, you will see your mediacenter shutting down on the TV within the few seconds. Then the power is cut about 40 seconds later. If the OS does not shut down first before the power is cut, then you need to double check your script installation.

OpenElec (tested with OpenElec 3.2.2, 3.2.3, 3.2.4)

- Note : In OpenElec it is currently NOT safe to just call the poweroff command from the script. This would corrupt your SD card eventually. The script below mimics the proper shutdown sequence which is executed when you choose shutdown from the GUI.
- Use PuTTY to connect to your OpenElec installation, default user name is root, password openelec
- Key in :

  ```
  cd .config
  nano autostart.sh
  ```

- Copy and paste the following into the nano editor window. If there is already something in the file, just add the following as the last lines

  ```
  #!/bin/bash
  (
  /storage/.config/irswitch.sh
  )&
  ```

- Press ctrl+x to exit the editor, y to confirm, enter to save the file
- Key in

  ```
  chmod +x autostart.sh
  nano irswitch.sh
  ```

- Copy and paste the following text into the editor window

  ```
  #!/bin/bash
  # prevent restarting XBMC at shutdown
  LOCKDIR="/var/lock/"
  LOCKFILE="xbmc.disabled"
  # this is the GPIO pin receiving the shut-down signal
  GPIOpin1=22
  # functions
    add_omit_pids() {
      omit_pids="$omit_pids -o $1"
    }
    safe_shutdown () {
      touch "$LOCKDIR/$LOCKFILE"
      add_omit_pids $(pidof connmand)
      add_omit_pids $(pidof dbus-daemon)
  ```

```
    killall5 -15 $omit_pids
    for seq in `seq 1 10` ; do
      usleep 500000
      clear > /dev/tty1
      killall5 -18 $omit_pids || break
    done
    sync
    umount -a >/dev/null 2>&1
    poweroff -f
  }

echo "$GPIOpin1" > /sys/class/gpio/export
echo "in" > /sys/class/gpio/gpio$GPIOpin1/direction
while [ 1 = 1 ]; do
 sleep 1
 power=$(cat /sys/class/gpio/gpio$GPIOpin1/value)
 if [ $power != 0 ]; then
  sleep 5
  safe_shutdown
 fi
done
```

- Press ctrl+x to exit, y to confirm, enter to save the file
- Key in

  ```
  chmod +x irswitch.sh
  ```

- Reboot from the OpenElec GUI (do NOT use reboot or poweroff in the ssh session as this may corrupt your SD card).
- After reboot you can use the RemotePi Board to power cycle OpenElec

Raspbmc (tested with Raspbmc Built 20130925)

- Use PuTTY to connect to your Raspbmc installation, default user name is pi, password raspberry
- Key in :

  ```
  cd /etc
  sudo nano rc.local
  ```

- Copy and paste the following into the nano editor window, just above the last line (before the line exit 0)

  ```
  /etc/irswitch.sh
  ```

- Press ctrl+x to exit the editor, y to confirm, enter to save the file
- Key in

  ```
  sudo nano irswitch.sh
  ```

- Copy and paste the following text into the editor window

  ```
  #!/bin/bash
  # this is the GPIO pin receiving the shut-down signal
  GPIOpin1=22
  echo "$GPIOpin1" > /sys/class/gpio/export
  echo "in" > /sys/class/gpio/gpio$GPIOpin1/direction
  while [ 1 = 1 ]; do
   sleep 1
   power=$(cat /sys/class/gpio/gpio$GPIOpin1/value)
   if [ $power != 0 ]; then
    sudo initctl stop xbmc
    sleep 5
    sudo shutdown -h now
  ```

```
 fi
done
```

- Press ctrl+x to exit, y to confirm, enter to save the file
- Key in

  ```
  sudo chmod +x irswitch.sh
  sudo shutdown –r now
  ```

- After reboot you can use the RemotePi Board to power cycle Raspbmc

XBian (tested with XBian 1.0 Beta 2)

- Use PuTTY to connect to your XBian installation, default user name is xbian, password raspberry
- Key in :

  ```
  cd /etc
  sudo nano rc.local
  ```

- Copy and paste the following into the nano editor window, just above the last line (before the line exit 0)

  ```
  /etc/irswitch.sh
  ```

- Press ctrl+x to exit the editor, y to confirm, enter to save the file
- Key in

  ```
  sudo nano irswitch.sh
  ```

- Copy and paste the following text into the editor window

  ```
  #!/bin/bash
  # this is the GPIO pin receiving the shut-down signal
  GPIOpin1=22
  echo "$GPIOpin1" > /sys/class/gpio/export
  echo "in" > /sys/class/gpio/gpio$GPIOpin1/direction
  ```

```
while [ 1 = 1 ]; do
 sleep 1
 power=$(cat /sys/class/gpio/gpio$GPIOpin1/value)
 if [ $power != 0 ]; then
  sudo poweroff
 fi
done
```

- Press ctrl+x to exit, y to confirm, enter to save the file
- Key in

```
sudo chmod +x irswitch.sh
sudo reboot
```

- After reboot you can use the RemotePi Board to power cycle XBian

RasPlex (tested with RasPlex 0.3.1)

- Use PuTTY to connect to your RasPlex installation, default user name is root, password rasplex
- The rest of the configuration for RasPlex is the same as for OpenElec, see above. (RasPlex is based on OpenElec)

For the newest version of these instructions please go to :

https://msl-digital-solutions.myshopify.com/pages/more-information