/etc/rc.local

```
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the
execution
# bits.
#
# By default this script does nothing.

# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
  printf "My IP address is %s\n" "$_IP"
fi

# Initialize LCD in red200 mode
(~/ctproject/adafruit_scripts/redbackground200.py)
/usr/bin/sudo /usr/bin/python
/home/pi/ctproject/adafruit_scripts/lcdstartup.py /dev/ttyACM0

# Run python script to listen for "Next" button press
# GPIO pin 35
/usr/bin/sudo /usr/bin/python
/home/pi/ctproject/audio_scripts/29jul2015/next.py &

# Run python script to listen for "Shutdown" button press
# GPIO pin 37
/usr/bin/sudo /usr/bin/python
/home/pi/ctproject/audio_scripts/29jul2015/shutdown.py &

# Run main python script to listen for button presses
# GPIO pins 16 and 18
/usr/bin/sudo /usr/bin/python
/home/pi/ctproject/audio_scripts/29jul2015/audioPlaylist.py &

exit 0
```

displayon.py

```
import serial
import sys
import time

# 16x2 LCD:
ROWS = 2
COLS = 16

def matrixwritecommand(commandlist):
    commandlist.insert(0, 0xFE)
    #ser.write(bytearray([0xFE]))
    #time.sleep(0.1);
    for i in range(0, len(commandlist)):
         #print chr(commandlist[i]),
         ser.write(chr(commandlist[i]))
    #ser.write(bytearray(commandlist))

# 1. get serial port
if len(sys.argv) != 2:
    print "Usage: python test.py <serialport>"
    exit(0)

ser = serial.Serial(sys.argv[1], 9600, timeout=1)
matrixwritecommand([0x58])

# set size
matrixwritecommand([0xD1, COLS, ROWS]);
matrixwritecommand([0x58])

# turn on display
ser.write("Display on");
matrixwritecommand([0x42, 0])
time.sleep(0.3);
```

displayoff.py

```
import serial
import sys
import time

# 16x2 LCD:
ROWS = 2
COLS = 16

def matrixwritecommand(commandlist):
    commandlist.insert(0, 0xFE)
    #ser.write(bytearray([0xFE]))
    #time.sleep(0.1);
    for i in range(0, len(commandlist)):
         #print chr(commandlist[i]),
         ser.write(chr(commandlist[i]))
    #ser.write(bytearray(commandlist))

# 1. get serial port
if len(sys.argv) != 2:
    print "Usage: python test.py <serialport>"
    exit(0)

ser = serial.Serial(sys.argv[1], 9600, timeout=1)
matrixwritecommand([0x58])

# set size
matrixwritecommand([0xD1, COLS, ROWS]);
matrixwritecommand([0x58])

# turn off display
#ser.write("off");
matrixwritecommand([0x46])
time.sleep(0.3);
```

redbackground200.py

```
import serial
import sys
import time

# 16x2 LCD:
ROWS = 2
COLS = 16

def matrixwritecommand(commandlist):
    commandlist.insert(0, 0xFE)
    #ser.write(bytearray([0xFE]))
    #time.sleep(0.1);
    for i in range(0, len(commandlist)):
         #print chr(commandlist[i]),
         ser.write(chr(commandlist[i]))
    #ser.write(bytearray(commandlist))

# 1. get serial port
if len(sys.argv) != 2:
    print "Usage: python test.py <serialport>"
    exit(0)

ser = serial.Serial(sys.argv[1], 9600, timeout=1)
matrixwritecommand([0x58])

# set size
matrixwritecommand([0xD1, COLS, ROWS]);
matrixwritecommand([0x58])

# set color
matrixwritecommand([0xD0, 200, 0, 0])
time.sleep(0.005)
```

peacefuldays.py

```
import serial
import sys
import time

# 16x2 LCD:
ROWS = 2
COLS = 16

def matrixwritecommand(commandlist):
    commandlist.insert(0, 0xFE)
    #ser.write(bytearray([0xFE]))
    #time.sleep(0.1);
    for i in range(0, len(commandlist)):
         #print chr(commandlist[i]),
         ser.write(chr(commandlist[i]))
    #ser.write(bytearray(commandlist))

# 1. get serial port
if len(sys.argv) != 2:
    print "Usage: python test.py <serialport>"
    exit(0)

ser = serial.Serial(sys.argv[1], 9600, timeout=1)
matrixwritecommand([0x58])

# set size
matrixwritecommand([0xD1, COLS, ROWS]);
matrixwritecommand([0x58])

# set color
matrixwritecommand([0xD0, 200, 0, 0])
time.sleep(0.005)

# Write text to screen
# New line begins...      X               X
ser.write("Playing         Peaceful Days");
time.sleep(4)

#matrixwritecommand([0x58])
```

audioPlaylist.py

```
import os
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BOARD)
GPIO.setup(16, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)
GPIO.setup(18, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)
#GPIO.setup(35, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)
#GPIO.setup(37, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)

while True:
      if(GPIO.input(16) == 1):
            os.system('sudo python
/home/pi/ctproject/adafruit_scripts/playlist1.py /dev/ttyACM0')
            os.system('sudo python
/home/pi/ctproject/adafruit_scripts/apremonition.py /dev/ttyACM0 &')
            os.system('mpg123 -q -m /home/pi/ctproject/media/Chrono\
Trigger/Disc\ 1/01.\ A\ Premonition.mp3')
            os.system('sudo python
/home/pi/ctproject/adafruit_scripts/chronotrigger.py /dev/ttyACM0')
            os.system('mpg123 -q -m /home/pi/ctproject/media/Chrono\
Trigger/Disc\ 1/02.\ Chrono\ Trigger.mp3')
            os.system('sudo python
/home/pi/ctproject/adafruit_scripts/displayoff.py /dev/ttyACM0')
            #time.sleep(3.0)

      if(GPIO.input(18) == 1):
            os.system('sudo python
/home/pi/ctproject/adafruit_scripts/morningsunlight.py
/dev/ttyACM0')
            os.system('mpg123 -q -m /home/pi/ctproject/media/Chrono\
Trigger/Disc\ 1/03.\ Morning\ Sunlight.mp3 ')
            os.system('sudo python
/home/pi/ctproject/adafruit_scripts/peacefuldays.py /dev/ttyACM0')
                  os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 1/04.\ Peaceful\
Days.mp3 ')
            os.system('sudo python
/home/pi/ctproject/adafruit_scripts/guardiamillenialfair.py
/dev/ttyACM0')
                  os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 1/06.\ Guardia\
Millenial\ Fair.mp3 ')
            os.system('sudo python
/home/pi/ctproject/adafruit_scripts/memoriesofgreen.py
/dev/ttyACM0')
                  os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 1/05.\ Memories\ Of\
Green.mp3 ')
            os.system('sudo python
/home/pi/ctproject/adafruit_scripts/600AD.py /dev/ttyACM0')
```

```
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 1/09.\ Wind\
Scene.mp3 ')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/secretoftheforest.py
/dev/ttyACM0')
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 1/11.\ Secret\ Of\
The\ Forest.mp3 ')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/thetrial.py /dev/ttyACM0')
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 1/21.\ The\ Trial.mp3
')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/thebrinkoftime.py /dev/ttyACM0')
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 2/13.\ The\ Brink\
Of\ Time.mp3 ')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/delightfulspekkio.py
/dev/ttyACM0')
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 2/14.\ Delightful\
Spekkio.mp3 ')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/undergroundsewer.py
/dev/ttyACM0')
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 2/16.\ Underground\
Sewer.mp3 ')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/rhythmofwind.py /dev/ttyACM0')
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 2/20.\ Rhythm\ Of\
Wind\ Sky\ \&\ Earth.mp3 ')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/burnbobonga.py /dev/ttyACM0')
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 2/21.\ Burn\!\
Bobonga\!.mp3 ')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/singingmountain.py
/dev/ttyACM0')
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 3/01.\ Singing\
Mountain\ \(Unreleased\).mp3 ')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/timecircuits.py /dev/ttyACM0')
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 3/04.\ Time\
Circuits.mp3 ')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/underseapalace.py /dev/ttyACM0')
```

```
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 3/08.\ Undersea\
Palace.mp3 ')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/chronoandmarle.py /dev/ttyACM0')
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 3/09.\ Chrono\ And\
Marle.mp3 ')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/epilogue.py /dev/ttyACM0')
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 3/16.\ Epilogue\ -\
To\ Good\ Friends.mp3 ')
                os.system('sudo python
/home/pi/ctproject/adafruit_scripts/tofarandawaytimes.py
/dev/ttyACM0')
                os.system('mpg123 -q -m
/home/pi/ctproject/media/Chrono\ Trigger/Disc\ 3/17.\ To\ Far\ Away\
Times.mp3 ')
            os.system('sudo python
/home/pi/ctproject/adafruit_scripts/thebrinkoftime_acidjazz.py
/dev/ttyACM0')
                os.system('mpg123 -q -m -@
/home/pi/ctproject/media/Chrono\ Trigger\ -\ The\ Brink\ Of\
Time/TheBrinkOfTime.m3u ')
            os.system('sudo python
/home/pi/ctproject/adafruit_scripts/displayoff.py /dev/ttyACM0')
            #time.sleep(3.0)

      #if(GPIO.input(35) == 1):
            #os.system('sudo pkill -9 mpg123')
            #time.sleep(2.0)

      #if(GPIO.input(37) == 1):
            #os.system('sudo shutdown -h -P now')
            #time.sleep(3.0)
GPIO.cleanup()
```

next.py

```
import os
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BOARD)
GPIO.setup(35, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)

while True:
        if(GPIO.input(35) == 1):
           os.system('sudo pkill -9 mpg123')
           #time.sleep(6.0)
           #os.system('sudo pkill -9 mpg123')

GPIO.cleanup()
```

shutdown.py

```
import os
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BOARD)
GPIO.setup(37, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)

while True:
        if(GPIO.input(37) == 1):
           os.system('sudo shutdown -h -P now')
           time.sleep(2.0)

GPIO.cleanup()
```