Kodi NUC main setup

BIOS SETTINGS

Enter bios:

```
systemctl reboot --firmware-setup
```

Set SGX to Enabled in BIOS to avoid boot message relating to it. (can possibly be set by software https://github.com/intel/sgx-software-enable) Set Fan control to custom with fanless at 48C, min 75, 30% at 1C, min 65 for RAM.

Suggested Custom Fan Control Mode: Primary Temperature Sensor to Processor, Fan Off Capability - enabled at 48C, Minimum Temperature to 85, Minimum Duty Cycle to 40, Duty Cycle Increment to 6 Secondary Temperature Sensor to Memory, Minimum Temperature to 60, Minimum Duty Cycle to 40, Duty Cycle Increment to 6.

Set CEC to desired settings

KODI INSTALL

Base OS install of Ubuntu server LTS 22.04.3 (orig notes from 20.04.1)

1. Old Ubuntu server releases Current Ubuntu server releases
2. install default server (not minimized)
3. Do not use lvm and/or encryption.
4. setup default user 'kodi' with password kodi (or setup your own user and add the kodi user later)
5. install no snap packages
6. install openssh
7. trigger reboot

copy ssh-keys manually to /home/kodi/.ssh/authorized_keys or use ssh-copy-id kodi@ip-address then proceed with configuration through remote ssh terminal.

```
sudo apt-get update
sudo apt-get install software-properties-common xorg xserver-xorg-legacy
alsa-utils mesa-utils git-core librtmp1 libmad0 lm-sensors libmpeg2-4 avahi-
daemon libnfs13 libva2 vainfo i965-va-driver linux-firmware dbus-x11 openbox
pastebinit xserver-xorg-video-intel alsa
sudo apt-get dist-upgrade
```

Support NFS mounts on system level:

```
sudo apt-get install nfs-common
```

Some house-cleaning to begin with:

```
#remove snapd
sudo apt purge snapd ubuntu-core-launcher squashfs-tools

#remove lvm2 (unless system uses LVM) Note: may remove ubuntu-server package
for cleanup
sudo apt remove lvm2 --purge
#remove raid support
sudo apt-get purge mdadm

#remove iscsi (IP based scsi support for remote storage)
sudo apt-get purge open-iscsi
#Remove udisks2: desktop environment option to mount/umount
sudo apt-get purge udisks2
#remove btrfs packages
sudo apt-get purge btrfs-progs
#remove iptables
sudo apt-get purge iptables
#remove wireless support
sudo apt-get purge wpasupplicant
#remove Open VM tools as not running in a VM
sudo apt-get purge open-vm-tools

sudo apt-get purge cloud-init* cloud-guest-utils

sudo apt-get purge ubuntu-advantage-tools
sudo apt-get purge cryptsetup*



#install zip and unzip
sudo apt-get install zip unzip samba-libs

# if the latest kernel is required or desired, install the hardware
enablement kernel (currently 5.11.x)
#sudo apt-get install --install-recommends linux-generic-hwe-20.04
```

Disable IPv6 support unless required:

```
echo "# Disable IPv6" | sudo tee /etc/sysctl.conf
echo "net.ipv6.conf.all.disable_ipv6 = 1" | sudo tee /etc/sysctl.conf
echo "#" | sudo tee /etc/sysctl.conf
sysctl -p
```

Remove the Ubuntu MOTD text on login and it's auto update

```
sed -i "s/ENABLED=1/ENABLED=0/" /etc/default/motd-news # run as root
sudo systemctl disable motd-news
sudo systemctl disable motd-news.timer
```

```
apt-get remove --purge landscape-common
sudo chmod -R 0644 /etc/update-motd.d/
sudo vi /etc/default/motd-news
#set enabled=0
sudo vi /etc/pam.d/login
sudo vi /etc/pam.d/sshd
#comment out these lines:
#session    optional    pam_motd.so motd=/run/motd.dynamic
#session    optional    pam_motd.so noupdate


# extreme version doesn't even show last login info
#sudo touch ~/.hushlogin
```

Allow “everyone” to start the x server

```
sudo dpkg-reconfigure xserver-xorg-legacy
echo "needs_root_rights=yes" | sudo tee /etc/X11/Xwrapper.config
```

Create the kodi user and it add it the relevant groups. If you have created the kodi user during installation only do the usermod part.

```
sudo adduser kodi
sudo usermod -a -G cdrom,audio,video,plugdev,users,dialout,dip,input kodi
```

Permission to shutdown, suspend the computer, therefore create the file /etc/polkit-1/localauthority/50-local.d/custom-actions.pkla with the following content:

```
[Actions for kodi user]
Identity=unix-user:kodi
Action=org.freedesktop.login1.*;org.freedesktop.udisks2.*
ResultAny=yes
ResultInactive=yes
ResultActive=yes

[Untrusted Upgrade]
Identity=unix-user:kodi
Action=org.debian.apt.upgrade-packages;org.debian.apt.update-cache
ResultAny=yes
ResultInactive=yes
ResultActive=yes
```

We need a simple systemd service file (this one actively waits on network connection, see: network-online.target remove that if you don't need to wait) Create the following file and put the listing into it: /etc/systemd/system/kodi.service

```
[Unit]
Description = kodi-standalone using xinit
Requires = dbus.service
After = systemd-user-sessions.service sound.target network-online.target
```

```
mysql.service
Wants = mysql.service

[Service]
User = kodi
Group = kodi
Type = simple
PAMName=login
ExecStart = /usr/bin/xinit /usr/bin/dbus-launch --exit-with-session
/usr/bin/kodi-standalone -- :0 -nolisten tcp vt7
Restart = always
RestartSec = 5
#ExecStart = /usr/bin/xinit /usr/bin/dbus-launch --exit-with-session
/usr/bin/openbox-session -- :0 -nolisten tcp vt7
#Restart = on-abort

[Install]
WantedBy = multi-user.target
```

edit /etc/security/limits.conf and add before the end. remember kodi is the username, not the application. This will allow your user to get the audio thread a bit more priority.

```
echo "kodi             -       nice            -1" | sudo tee
/etc/security/limits.conf
```

Fake display-manager.service to not make plymouth or something else complain.

```
sudo ln -s /etc/systemd/system/kodi.service /etc/systemd/system/display-
manager.service
```

```
#check ubuntu version
lsb_release -a
```

Add the stable or nightly repository and install kodi:

```
#stable 20
sudo add-apt-repository ppa:team-xbmc/ppa
sudo apt-get update
sudo apt-get dist-upgrade
sudo apt-get install kodi kodi-x11 kodi-inputstream-adaptive kodi-vfs-
libarchive kodi-vfs-rar

#nightly 21 - combined x11/wayland/gbm version as of nightly 28/Oct/2020
sudo add-apt-repository ppa:team-xbmc/xbmc-nightly
sudo apt-get update
sudo apt-get dist-upgrade
sudo apt-get install kodi kodi-bin kodi-inputstream-adaptive kodi-vfs-
libarchive kodi-vfs-rar

#remove joystick support unless desired for retro-gaming
```

```
sudo apt-get purge kodi-peripheral-joystick
```

As we use openbox as our display manager, we need to auto start kodi, therefore create:

```
sudo mkdir -p /home/kodi/.config/openbox
sudo touch /home/kodi/.config/openbox/autostart
sudo chown kodi:kodi /home/kodi/.config -R
```

now we write the following into the created /home/. kodi/.config/openbox/autostart file, this will automatically switch your TV to full range (please copy the lines, don't try to type the '` and so on, this code only works for one (1) connected TV, if you have multiple devices extend it to a loop):

```
OUTPUT=`xrandr -display :0 -q | sed '/ connected/!d;s/ .*//;q'`
xrandr -display :0 --output $OUTPUT --set "Broadcast RGB" "Full"
xsetroot #000000
xset s off -dpms
exec /usr/bin/kodi --standalone
while [ $? -ne 0 ]; do
exec /usr/bin/kodi --standalone
done
openbox --exit
```

Tweak the config further as per descriptions below depending on requirements (bluetooth, sound, boot logo, mysql database connection etc), otherwise start kodi:

```
sudo systemctl start kodi
```

Kodi Settings for Hardware decoding (VAAPI)

```
System ->Player->Processing:
Enable HQ Scalers for scaling above: 20%
Allow hardware acceleration - VAAPI: on
Use Mpeg-2 VAAPI: Yes
Use Mpeg-4 VAAPI: if you like
UseVC-1 VAAPI: on
Use VP8 VAAPI: on (if your device supports it)
Use VP9 VAAPI: on (if your device supports it)
Use HEVC VAAPI: on (if your device supports it)

System -> Player -> Playback
Adjust Refreshrate to match video: On start / stop
Sync Playback to Display: On if you don't use passthrough and Off if
passthrough enabled (* This makes no sense when you want to use passthrough,
why? Read: 2107461 (post)). In current versions (Jarvis) passthrough is
disabled by us if users wants to Sync Playback to Display to care for people
that refuse to read.
Adjust display refresh rate to match video: On Start / Stop

While watching a SD(!) video, that is accelerated by VAAPI, e.g. mpeg-2 or
```

```
h264, click the film role and choose: Deinterlacing-Method: VAAPI-MCDI or
VAAPI-MADI (Sandybridge) and VAAPI-BOB (BYT), Scaling Method: Lanczos3
Optimized and choose save for all files. Remember to do this only in
combination with the above "scaling above" for 20%. This Lanczos3 Optimized
filter is too heavy for BYTs, here you might - depending on the file -
choose Bilinear. (Hint the Deinterlace: Auto setting was removed some years
ago - we can now properly detect interlace / non interlaced content).

It is obviously clear, that you won't see the VAAPI-MCDI settings when you
play a video that is software accelerated only.
```

Possible ~/.kodi/userdata/advancedsettings.xml file for special needs:

```
You might need to create and advancedsettings.xml file as follows if you
have special needs:
<advancedsettings>
  <loglevel hide="false">0</loglevel>
 <cputempcommand>sensors|sed -ne "s/Core 0: \+[-+]\([0-9]\+\).*/\1
C/p"</cputempcommand>
 <gui>
   <algorithmdirtyregions>3</algorithmdirtyregions>
   <nofliptimeout>0</nofliptimeout>
 </gui>
<video>
 <latency>
   <delay>50</delay>
   <refresh>
     <min>23</min>
     <max>24</max>
     <delay>175</delay> <!-- set to zero or adjust if audio seems out of
sync with 24p movies -->
    </refresh>
 </latency>
</video>
</advancedsettings>
```

Color Management (only correct with above xrandr forced to FULL)

```
If your TV is limited range. Go to System -> Video Output and choose "Use
Limited Range", disable Dithering.
If your TV is full range. Go to System -> Video Output and unselect "Use
Limited Range" and enable Dithering with 8 bits.
If your projector is of low quality, use a dithering setting of 6 or 7 bits.

Remember the above settings only make sense, when you output in Full Range.
With the default xrandr setting of Limited 16:235 - those settings would be
totally wrong.
```

Disable output during boot

```
sudo vi /etc/default/grub
```

adjust the GRUB_CMDLINE_LINUX line to:

```
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash"
```

commit the change using

```
sudo update-grub
```

Install animated kodi logo bootup theme:

```
git clone https://github.com/solbero/plymouth-theme-kodi-animated-logo
cd plymouth-theme-kodi-animated-logo
sudo apt-get install fakeroot fonts-ubuntu plymouth-label
./build.sh
sudo dpkg -i plymouth-theme-kodi-animated-logo.deb
```

Enable graphical grub and store debug log as /var/log/plymouth-debug.log:

```
sudo vi /etc/default/grub
```

```
GRUB_GFXPAYLOAD_LINUX=auto
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash
plymouth:debug=file:/var/log/plymouth-debug.log"
# plymouth:force-drm
```

```
echo FRAMEBUFFER=y | sudo tee /etc/initramfs-tools/conf.d/framebuffer &&
sudo update-initramfs -u
```

commit the change using

```
sudo update-grub
```

prevent update-grub from scanning all drives other than the boot drive for linux installations including usb sticks

```
sudo vi /etc/default/grub
```

```
GRUB_DISABLE_OS_PROBER=true
```

commit the change using

```
sudo update-grub
```

Boot without connected TV / AVR

The Extended Display Identification Data (EDID) is data that describes the capabilities of a digital display (DVI,HDMI, Displayport, etc) and E-EDID contains the audio capabilities as well.

NOTE: Kernel 4.4+ is required or audio EDID information won't be copied. Verify with

```
uname -a
```

First find the connected port (HDMI/Displayport):

```
ls -la /sys/class/drm
cat /sys/class/drm/card0-DP-1/status
connected
```

edit /etc/default/grub and adjust the GRUB_CMDLINE_LINUX_DEFAULT with your port details and desired display resolution and refresh rate. Add D directly after the refresh rate for Digital output! Note the port info here does not start with “card0-”

```
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash video=HDMI-A-1:1920x1080@50D
drm_kms_helper.edid_firmware=HDMI-A-1:edid/edid.bin"
```

Make sure the system was booted connected to the AVR/TV and everything is working. Then proceed to copy the EDID:

```
sudo mkdir -p /lib/firmware/edid
sudo cp /sys/class/drm/card0-HDMI-A-1/edid /lib/firmware/edid/edid.bin
```

Now we need to make a little hack and also convince ubuntu to load this edid.bin in initramfs. Create /etc/initramfs-tools/hooks/include-edid-data file and write into it:

/etc/initramfs-tools/hooks/include-edid-data

```
#!/bin/sh

PREREQ="udev"
prereqs()
{
   echo "$PREREQ"
}

case $1 in
prereqs)
   prereqs
   exit 0
   ;;
esac

. /usr/share/initramfs-tools/hook-functions
```

```
# Begin real processing below this line

if [ ! -e "${DESTDIR}/lib/firmware/edid" ]; then
    mkdir -p "${DESTDIR}/lib/firmware/edid"
fi

if [ -r "/lib/firmware/edid/edid.bin" ]; then
   cp "/lib/firmware/edid/edid.bin" "${DESTDIR}/lib/firmware/edid/"
fi

manual_add_modules i915 radeon
exit 0
```

Make it executable

```
sudo chmod +x /etc/initramfs-tools/hooks/include-edid-data
```

Finish with

```
sudo update-initramfs -u
sudo update-grub
sudo reboot
```

ALSA SOUND

First make sure Pulseaudio is not installed or if it is, uninstall it. If there's a reference to `snd_hda_intel` in the output, then the kernel module has already been loaded as the driver:

```
sudo lsmod | grep snd_hda_intel
```

This command will list all sound devices. You'll need this to identify which sound device you wish to use later and to determine if there are multiple devices or just one.

```
sudo aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: PCH [HDA Intel PCH], device 0: ALC283 Analog [ALC283 Analog]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: PCH [HDA Intel PCH], device 1: ALC283 Digital [ALC283 Digital]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: PCH [HDA Intel PCH], device 3: HDMI 0 [HDMI 0]
  Subdevices: 0/1
  Subdevice #0: subdevice #0
card 0: PCH [HDA Intel PCH], device 7: HDMI 1 [HDMI 1]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: PCH [HDA Intel PCH], device 8: HDMI 2 [HDMI 2]
```

```
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: PCH [HDA Intel PCH], device 9: HDMI 3 [HDMI 3]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 0: PCH [HDA Intel PCH], device 10: HDMI 4 [HDMI 4]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

This command will show your model and codec chip to get further help via google in case of issues.

```
cat /proc/asound/card*/codec* | grep Codec
```

These two commands will show the current driver options for the sound models:

```
zless /usr/share/doc/alsa-base/driver/HD-Audio-Models.txt.gz
zless /usr/share/doc/alsa-base/driver/ALSA-Configuration.txt.gz
```

```
echo "options snd-hda-intel enable_msi=1 bdl_pos_adj=1,48" | sudo tee
/etc/modprobe.d/alsa-base.conf
```

Apply the changes and reload alsa by rebooting or using:

```
sudo alsa force-reload
```

Monitor issues in dmesg

```
sudo dmesg
```

Further references:
https://forum.kodi.tv/showthread.php?tid=83614
https://wiki.gentoo.org/wiki/ALSA#A.2F52.2C_AC3.2C_Dolby.2C_DTS
https://forum.voidlinux.eu/t/solved-setting-up-alsa-sound-how-to/1416/20

WIFI

dmesg | grep iwl if firmware is not loaded at all, go to:
https://www.intel.com/content/www/us/en/support/network-and-i-o/wireless-networking/000005511.html copy the link of the relevant firmware to clipboard, download it with wget, untar , copy to /lib/firmware

Bluetooth

Apparently 2nd generation AFTV remotes use wifi instead of bluetooth.
The 1st gen AFTV remotes use bluetooth. They identify as

```
v1949:0401
  "[CHG] Device 5C:31:3E:80:FD:48 Modalias: bluetooth:v1949p0401d0112"
```

Fire TV stick remotes are from a different vendor and use v1949:0404
workarounds in this thread: https://forum.kodi.tv/showthread.php?tid=208890&page=7

```
sudo apt-get install bluez bluetooth
sudo /etc/init.d/networking restart
sudo /etc/init.d/bluetooth start
```

Add bluetooth device using command line:

```
hciconfig
sudo bluetoothctl
[NEW] Controller 30:E3:7A:D3:50:A3 nuc-kodi [default]
[bluetooth]# power on
Changing power on succeeded
[bluetooth]# devices
[bluetooth]# scan on
Discovery started
[CHG] Controller 30:E3:7A:D3:50:A3 Discovering: yes
[NEW] Device 5C:31:3E:80:FD:48 Amazon Fire TV Remote
[CHG] Device 5C:31:3E:80:FD:48 TxPower: 4
[CHG] Device 5C:31:3E:80:FD:48 Alias: 5C-31-3E-80-FD-48
[CHG] Device 5C:31:3E:80:FD:48 Name is nil
[bluetooth]# agent on
Agent registered
[bluetooth]# pair 5C:31:3E:80:FD:48
Attempting to pair with 5C:31:3E:80:FD:48
[CHG] Device 5C:31:3E:80:FD:48 Connected: yes
[CHG] Device 5C:31:3E:80:FD:48 Name: Amazon Fire TV Remote
[CHG] Device 5C:31:3E:80:FD:48 Alias: Amazon Fire TV Remote
[CHG] Device 5C:31:3E:80:FD:48 Modalias: bluetooth:v1949p0401d0112
[CHG] Device 5C:31:3E:80:FD:48 ServicesResolved: yes
[CHG] Device 5C:31:3E:80:FD:48 Paired: yes
Pairing successful
[CHG] Device 5C:31:3E:80:FD:48 ServicesResolved: no
[CHG] Device 5C:31:3E:80:FD:48 Connected: no
[CHG] Device 5C:31:3E:80:FD:48 Alias: 5C-31-3E-80-FD-48
[CHG] Device 5C:31:3E:80:FD:48 Name is nil
[bluetooth]# trust 5C:31:3E:80:FD:48
[CHG] Device 5C:31:3E:80:FD:48 Trusted: yes
Changing 5C:31:3E:80:FD:48 trust succeeded
[bluetooth]# pair 5C:31:3E:80:FD:48
Attempting to pair with 5C:31:3E:80:FD:48
Failed to pair: org.bluez.Error.AlreadyExists
[CHG] Device 5C:31:3E:80:FD:48 RSSI: -75
[bluetooth]# devices
Device 5C:31:3E:80:FD:48 5C-31-3E-80-FD-48
[bluetooth]# exit
```

```
Agent unregistered
[DEL] Controller 30:E3:7A:D3:50:A3 nuc-kodi [default]
sudo hcitool con
Connections:
sudo bluetoothctl -a
[NEW] Controller 30:E3:7A:D3:50:A3 nuc-kodi [default]
[NEW] Device 5C:31:3E:80:FD:48 5C-31-3E-80-FD-48
Agent registered
[CHG] Device 5C:31:3E:80:FD:48 Connected: yes
[CHG] Device 5C:31:3E:80:FD:48 Name: Amazon Fire TV Remote
[CHG] Device 5C:31:3E:80:FD:48 Alias: Amazon Fire TV Remote
[5C-31-3E-80-FD-48]# exit
```

CEC Configuration

BIOS Config:
https://www.intel.com/content/www/us/en/support/articles/000023500/mini-pcs/intel-nuc-kits.html

Possibly required:

```
sudo apt-get install cec-utils
```

Copy kodi configuration from Amazon Fire

wuff@gondor ~/Downloads/AFTV Kodi Backup $ adb connect 192.168.1.10 * daemon not running. starting it now on port 5037 * * daemon started successfully * connected to 192.168.1.10:5555 wuff@gondor ~/Downloads/AFTV Kodi Backup $ adb pull /sdcard/Android/data/org.xbmc.kodi/files/.kodi . pull: building file list... [.....] 31017 files pulled. 0 files skipped. 849 KB/s (2722037503 bytes in 3129.565s)

Setting up Kodi MySQL Database

http://kodi.wiki/view/MySQL/Setting_up_MySQL:

```
sudo apt-get install mysql-server
#create password when asked
sudo vi /etc/mysql/mysql.conf.d/mysqld.cnf
#change bind-address from 127.0.0.1 to actual IP (when used by others only)
or to 0.0.0.0 when used both internally and externally
sudo service mysql restart
mysql -u root -p
#enter password, then enter the following four commands in the console:
CREATE USER 'kodi' IDENTIFIED BY 'kodi';
GRANT ALL ON *.* TO 'kodi';
flush privileges;
```

```
exit
```

Setting up Kodi for use with MySQL (http://kodi.wiki/view/MySQL/Setting_up_Kodi): do NOT use locally mapped network shares for kodi as otherwise the db entries will refer to localhost which won't work on other devices. Use generic smb or nfs paths instead. Ideally using fixed IP addresses for the NAS and not hostnames.

If you use passwords on your network shares then you will need to copy (or sync) your passwords.xml file from the userdata folder to each Kodi device.

Export the video and music library as nfo files to export an existing library from a device. If it's a new setup, ignore this.

edit or add to /home/kodi/.kodi/userdata/advancedsettings.xml

```
<advancedsettings>
  <videodatabase>
    <type>mysql</type>
    <host>***.***.***.***</host>
    <port>3306</port>
    <user>kodi</user>
    <pass>kodi</pass>
  </videodatabase>
  <musicdatabase>
    <type>mysql</type>
    <host>***.***.***.***</host>
    <port>3306</port>
    <user>kodi</user>
    <pass>kodi</pass>
  </musicdatabase>
  <videolibrary>
    <importwatchedstate>true</importwatchedstate>
    <importresumepoint>true</importresumepoint>
  </videolibrary>
</advancedsettings>
```

transfer userdata/addon_data and /addons from old to new server. transfer Thumbnails if desired.

transfer: /home/kodi/.kodi/userdata/passwords.xml /home/kodi/.kodi/userdata/sources.xml /home/kodi/.kodi/userdata/mediasources.xml

sudo chown kodi:kodi /home/kodi/.kodi/addons/* -R sudo chown kodi:kodi /home/kodi/.kodi/userdata/addon_data/* -R

if amazon fire tv remote doesn't work properly - especially context menu key, create a custom keymap: sudo vi /home/kodi/.kodi/userdata/keymaps/gen.xml <keymap><global><keyboard><key id="61952">contextmenu</key></keyboard></global></keymap> then reboot

on kodi, go to the video sources and for each define the content and update the library. This will take a while.

Share Thumbnails on NAS drive

If multiple Kodi instances should share video/music thumbnails, the following lines need to be added to the advancedsettings.xml to redirect Kodi's special folders to a NAS drive: edit or add to /home/kodi/.kodi/userdata/advancedsettings.xml

```
<advancedsettings>
    <pathsubstitution>
      <substitute>
        <from>special://masterprofile/Thumbnails/Video/</from>
        <to>smb://192.168.1.10/xbmc/userdata/Thumbnails/Video</to>
      </substitute>
      <substitute>
        <from>special://masterprofile/Thumbnails/Music/</from>
        <to>smb://192.168.1.10/xbmc/userdata/Thumbnails/Video</to>
      </substitute>
    </pathsubstitution>
</advancedsettings>
```

Increase TV TTY font size

```
sudo dpkg-reconfigure console-setup
```

set to highest font-size in 4th step

Boblight / Arduino

On main linux pc:

```
sudo apt-get install arduino
```

start arduino, connect arduino via usb, select version and port in the IDE, upload desired sketch (program).

```
cd /tmp
wget https://www.tweaking4all.com/?wpfb_dl=78 -O t4a_boblight.ino
wget https://www.tweaking4all.com/?wpfb_dl=53 -O Adafruit_NeoPixel.zip
```

via sketch import library, import the Adafruit zip file
open the ino sketch, adjust to amount of leds, compile and upload it to the arduino

On KODI box:

```
sudo apt-get install build-essential libusb-1.0-0-dev libxrender-dev
libavcodec-dev libavformat-dev libswscale-dev libavdevice-dev
```

NOTE: original repository on google code doesn't seem to work any longer. use: https://github.com/bobo1on1/boblight

```
cd /tmp
git clone https://github.com/bobo1on1/boblight
cd boblight
sudo ./configure --without-portaudio --without-ffmpeg --without-x11 --
prefix=/usr
sudo ./configure --without-portaudio --without-opengl --without-x11 --
prefix=/usr
make && sudo make install
sudo cp arduino/duemilanove.conf /etc/boblight.conf

sudo boblightd
```

to install boblight config maker linux version, the following i386 libraries are required:

```
sudo apt-get install libatk1.0-0:i386 libcairo2:i386 libgdk-pixbuf2.0-0:i386
libpango-1.0-0:i386 libgtk2.0-0:i386
```

using `ldd Boblight\ Config\ Maker\ Linux` will show any other libraries that might be missing.

Bluray playback using MakeMKV

```
sudo apt-get install libaacs0 libbdplus0 libbluray2 libbluray-bdj libbluray-
bin

sudo add-apt-repository ppa:heyarje/makemkv-beta
sudo apt-get update
sudo apt-get install makemkv-bin makemkv-oss

cd /usr/lib/x86_64-linux-gnu/
sudo rm libaacs.so* libbdplus.so*
sudo ln -s libmmbd.so.0 libaacs.so.0
sudo ln -s libmmbd.so.0 libbdplus.so.0
```

```
sudo vi /etc/apt/preferences.d/libbdplus
```

```
Package: libbdplus*
Pin: version *
Pin-Priority: -1
```

```
sudo vi /etc/apt/preferences.d/libaacs
```

```
Package: libaacs*
Pin: version *
Pin-Priority: -1
```

#as kodi user paste beta key from http://www.makemkv.com/forum2/viewtopic.php?f=5&t=1053

```
mkdir ~/.MakeMKV
vi ~/.MakeMKV/settings.conf
```

```
app_Key="xxx"
```

Bash script for automated key update for cron:

updatemakemkvkey.sh

```
#!/bin/sh
url=$(curl -s "https://www.makemkv.com/forum/viewtopic.php?f=5&t=1053"
| grep "current beta key.*<\/code>" )
key=$(echo "$url" | sed -n 's/.*\(T-[0-9a-zA-Z]*\).*/\1/p')
echo "app_Key=\"$key\"" > ~/.MakeMKV/settings.conf
```

Python script for remote edit/update of DB entries

```
##https://github.com/MilhouseVH/texturecache.py
wget
https://raw.githubusercontent.com/MilhouseVH/texturecache.py/master/texturec
ache.py -O texturecache.py
chmod +x ./texturecache.py
```

examples:

```
./texturecache.py stats video  #shows stats
./texturecache.py remove movie 123  #removes movie 123
./texturecache.py J movies "Lion King"  #queries movies for entries 'Lion
King'
./texturecache.py missing movies "movies" #Lists missing movies from the
source 'movies'
./texturecache.py duplicates #List dulicate movies with the same IMDb number
./texturecache.py watched "mywatchlistbackup.txt" #Backup or restore movies
and tvshows watched status and restore points, to/from the specified text
file

  ascan      Scan entire audio library, or specific path
  vscan      Scan entire video library, or specific path
  aclean     Clean audio library
  vclean     Clean video library

  set        Set values on objects (movie, tvshow, episode, musicvideo,
album, artist, song) eg. "set movie 312 art.fanart
'http://assets.fanart.tv/fanart/movies/19908/hdmovielogo/zombieland-5145e97e
d73a4.png'"
```

```
  p          Display files present in texture cache that don't exist in the
media library
  P          Prune (automatically remove) cached items that don't exist in
the media library
  qa         Run QA check on movies, tags and tvshows, identifying media
with missing artwork or plots
  qax        Same as qa, but remove and rescan those media items with
missing details.
             Configure with qa.zero.*, qa.blank.* and qa.art.* properties.
Prefix field with ? to render warning only.
  sources    List all sources, or sources for specific media type (video,
music, pictures, files, programs) or label (eg. "My Movies")
```

Boblight settings via keyboard shortcut

https://github.com/bobo1on1/script.xbmc.boblight
put the file script.xbmc.boblight.keymap.xml into your “userdata/keymaps” folder.
You can access the settings during playback by hitting the key “b” or “B”. (Note: a former keymapping to that key will be overwritten by this keymap file. If there is a conflict you just can change the mapped key to something else).

script.xbmc.boblight.keymap.xml:

```
<keymap>
  <fullscreenvideo>
    <keyboard>
      <b>Addon.OpenSettings(script.xbmc.boblight)</b> <!-- shortcut key b to
boblight settings -->
      <B>Addon.OpenSettings(script.xbmc.boblight)</B> <!-- shortcut key B to
boblight settings -->
    </keyboard>
  </fullscreenvideo>
</keymap>
```

Sources

This article is based on the main instructions from here

Refresh Kodi skin:
reddit.com/r/kodi/comments/4ml9ll/refresh_skin_command/
forum.kodi.tv/showthread.php?tid=76233

scantokodi.sh command (to let transmission notify kodi of a finishe download so that it will scan that particular file to the library without doing a full library scan)
https://gist.github.com/zilexa/e82211bbfe41de4274b44d0b48b14642

#led
lifehacker.com/5433355/build-a-color-aware-backlight-for-your-hdtv-setup

www.tweaking4all.com/home-theatre/xbmc/xbmc-boblight-openelex-ws2811-ws2812/
www.tweaking4all.com/hardware/arduino/first-arduino-project
lektiondestages.blogspot.co.uk/2013/01/setting-up-arduino-ldp8806-ambilight.html?m=1
https://github.com/arduino/arduino-builder/blob/master/README.md
https://github.com/arduino/Arduino/blob/master/build/shared/manpage.adoc
https://kodi.tv/addon/services/kodi-boblight
github.com/bobo1on1/boblight
kodi.tv/addon-author/bobo1on1
forum.kodi.tv/showthread.php?tid=116331
github.com/bobo1on1/script.xbmc.boblight/
memphiz.f00-bar.net/dl/boblight
forum.kodi.tv/showthread.php?tid=101406
learn.adafruit.com/adalight-diy-ambient-tv-lighting/troubleshooting

#wifi direct
thangamaniarun.wordpress.com/2013/03/03/how-to-use-wi-fi-direct-on-androidubuntu-part1/
thangamaniarun.wordpress.com/2013/04/07/how-to-setup-wi-fi-direct-on-androidubuntu-terminal-part2/
thangamaniarun.wordpress.com/2014/03/30/how-to-setup-wi-fi-direct-on-androidubuntu-terminal-part3/