

RDK CommonPort

- 1. Build environment
- 2. Build Steps for rdk-next (Master)
- 3. Build Steps for a stable Feature branch (In-Progress)
- 4. Steps to build libloader-apps:
- 5. IPSTB Flashing Steps
 - 5.1.1. Locate the Image file from build directory
 - 5.2. How to flash X4-SC2 image
- 6. Boot up UI

1. Build environment

Please register to code.rdkcentral.com and share to RDKM to enable Amlogic repo access. (Need Amlogic to approve access to the request).

On build machine, add code.rdkcentral.com login in your netrc file.

ie vi ~/.netrc

- cat ~/.netrc
 - machine code.rdkcentral.com login XXXXXXXXXXXX password XXXXXXXXXXXX

Update ~/.ssh/config for Amlogic openlinux.

To access, Amlogic DRM, need to share ssh private key to Amlogic and get it enable for Openlinux access for Widevine, Playready, Secmem repos.

Configure it accordingly in ~/.ssh/config

- cat ~/.ssh/config
 - Host openlinux.amlogic.com
 - HostName openlinux.amlogic.com
 - IdentityFile ~/.ssh/id_rsa

2. Build Steps for rdk-next (Master)

- repo init -u <https://code.rdkcentral.com/r/collaboration/soc/amlogic/aml-accel-manifests> -b google-rdk-aml-collab -m rdkv-aml-open-va-google.xml
- repo sync -j4 --no-clone-bundle
- source ./meta-rdk-aml/set-env.sh mesons2-5.4-lib32-ah212 --use-rdk-ui --playready --widevine --rdk-starboard --enable-evergreen-full --enable-rialto --enable-webkit238 --enable-wper4 --ipclient
- bitbake lib32-rdk-ipstb-oss-image

Note: To enable build with DRM(Playready & Widevine), Please reach out to Amlogic for Openlinux access for DRM libraries. Need to have below repository access.

NOTE: Both Amlogic and RDKM teams enable MS12 audio on the build. To enable MS12 audio in the build, need to enable build option "--enable-dv-ms12" .

This may effect some of YTS tests with respect to partial audio to fail if no MS12 audio(though ms12 is not directly needed, due to audio path, it may need ms12 audio to enable).

Check if you have access to Amlogic restricted repos.

Execute below command to check if you have PlayReady access

```
git clone ssh://git@openlinux.amlogic.com/vendor/playready.git
```

Execute below command to check if you have Widevine access

```
git clone ssh://git@openlinux.amlogic.com/yocto/vendor/widevine
```

Please contact your Amlogic sales or FAE teams to get restricted DRM repo access if you do not have yet.

3. Build Steps for a stable Feature branch (In-Progress)

- repo init -u <https://code.rdkcentral.com/r/collaboration/soc/amlogic/aml-accel-manifests> -b google-rdk-aml-collab -m feature-rdkv-aml-open-vagoogle.xml
- repo sync -j4 --no-clone-bundle
- source ./meta-rdk-aml/set-env.sh mesons2-5.4-lib32-ah212 --use-rdk-ui --playready --widevine --rdk-starboard --enable-evergreen-full --enable-rialto --enable-webkit238 --enable-wper4 --ipclient
- bitbake lib32-rdk-ipstb-oss-image

Note: To enable build with DRM(Playready & Widevine), Please reach out to Amlogic for Openlinux access for DRM libraries. Need to have below repository access.

4. Steps to build libloader-apps:

- **Libloader-app recipe path:** meta-rdk-video/recipes-extended/cobalt/libloader-app_24.lts.stable.bb
- **Libloader-app append path:** meta-rdk-aml/recipes-extended/cobalt/libloader-app_24.lts.stable.bbappend
- **Cobalt plugin recipe path:** meta-rdk-video/recipes-extended/cobalt/cobalt-plugin_git.bb
- **RDk Starboard path:** build/tmp/work/armv7at2hf-neon-rdkmllib32-linux-gnueabi/lib32-libloader-app/24.lts.stable+gitAUTOINC+ecdcfdb8e5_8e7fcc2fe2-40/git/third_party/starboard/rdk/shared/
- To build libloader-app or cobalt-plugin modules individually:
 - bitbake lib32-libloader-app
 - bitbake lib32-cobalt-plugin
- To make changes in Starboard module:
 - Navigate to the starboard path - build/tmp/work/armv7at2hf-neon-rdkmllib32-linux-gnueabi/lib32-libloader-app/24.lts.stable+gitAUTOINC+ecdcfdb8e5_8e7fcc2fe2-40/git/third_party/starboard/rdk/shared/
 - Make necessary changes
 - Come back to build folder
 - Re-compile and build again - bitbake lib32-libloader-app -c compile && bitbake lib32-libloader-app

Cobalt library path:

- **Evergreen libloader-app library:** build/tmp/work/armv7at2hf-neon-rdkmllib32-linux-gnueabi/lib32-libloader-app/24.lts.stable+gitAUTOINC+ecdcfdb8e5_8e7fcc2fe2-40/image/usr/lib/libloader_app.so
- **Cobalt plugin library:** build/tmp/work/armv7at2hf-neon-rdkmllib32-linux-gnueabi/lib32-cobalt-plugin/git+gitAUTOINC+8e7fcc2fe2-r0/image/usr/lib/wpeframework/plugins/libWPEFrameworkCobaltImpl.so

5. IPSTB Flashing Steps

5.1.1. Locate the Image file from build directory

The reference device Image shall be created under \${BUILDDIR}/tmp/deploy/images/<machine>/aml_upgrade_package.img which can be used to flash the device using below methods.

1) Using Burning tool:

Download AMlogic Burning tool from the link [Amlogic Video Accelerator Collaboration](#)).

NOTE: Please reachout to Amlogic to get USB Burning tool if no access to the wiki page.

5.2. How to flash X4-SC2 image

Connect all necessary cables to AH212 box as below. It's easy to do with place AH212 up-side-down.

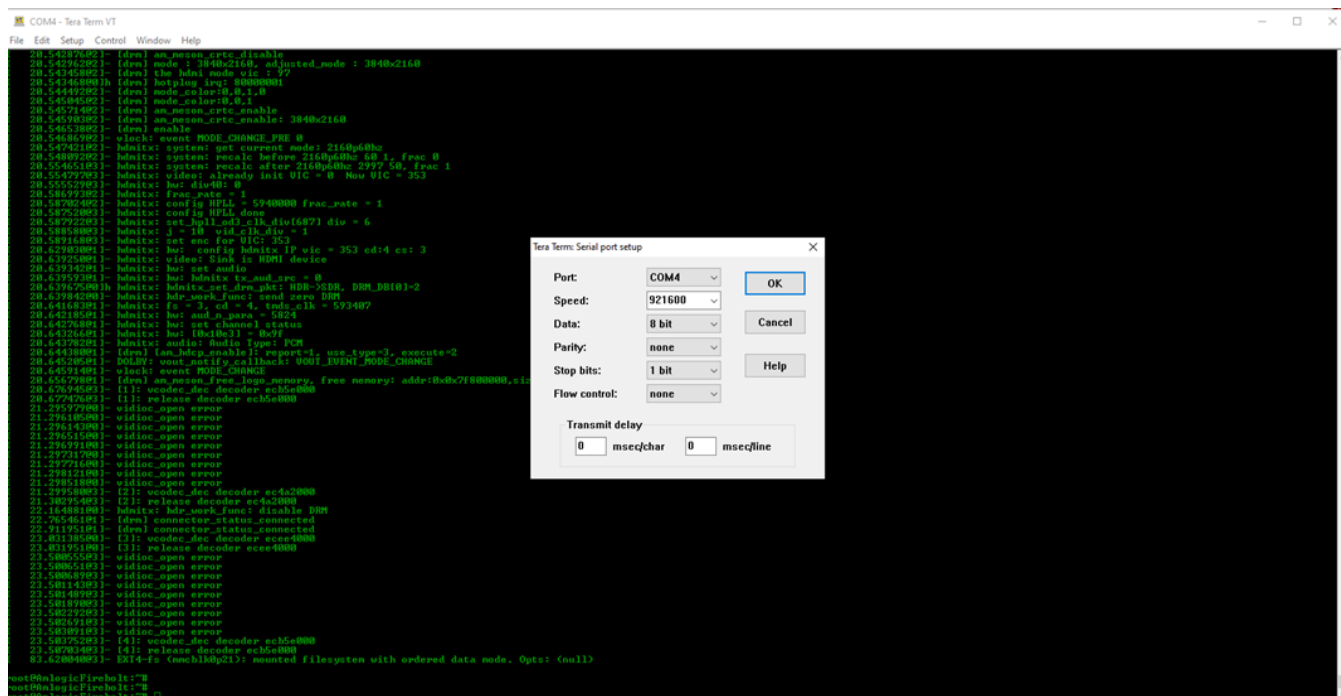
From left to right: Power, USB-C, Console, HDMI, Ethernet

USB-C is only using for flash image.

USB-Micro is for Console connection.



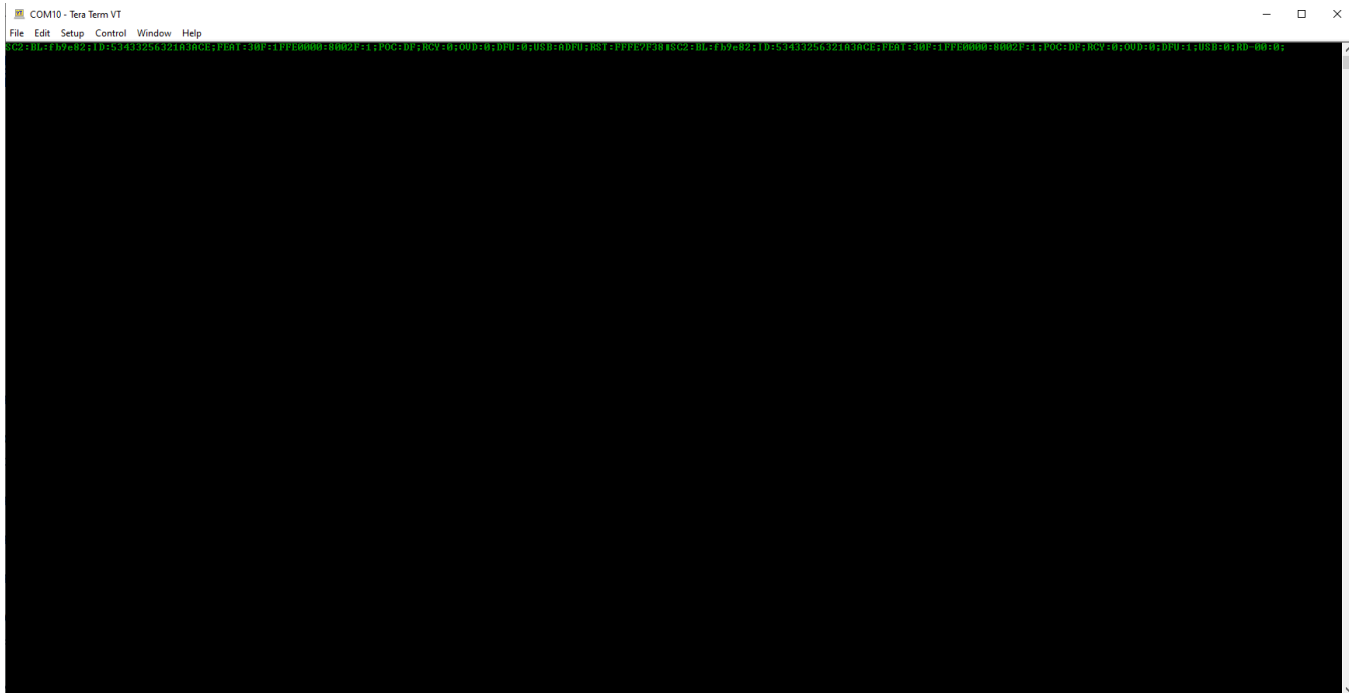
Setup the serial port settings (921600 baud rate 8N1)



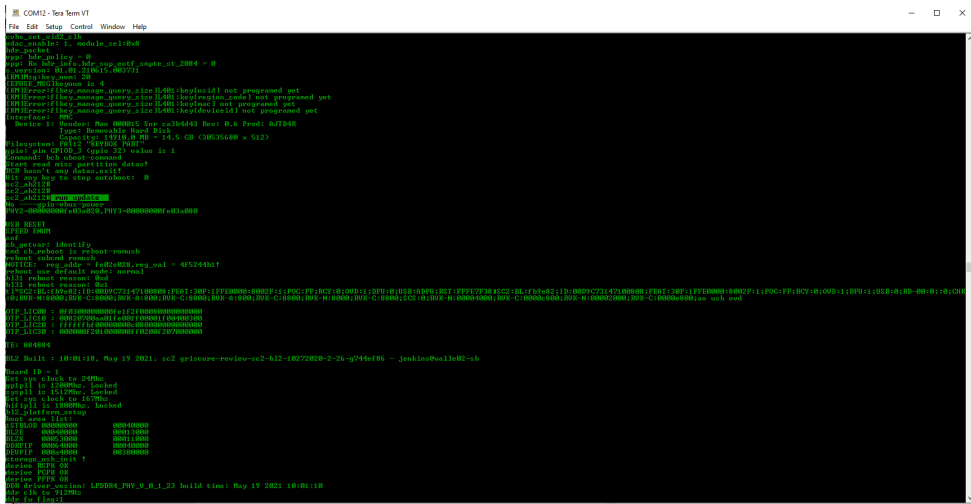
Option 1: Enable USB burning mode for V3 Buring Tool App by holding USB BOOT button and apply AC to box as below



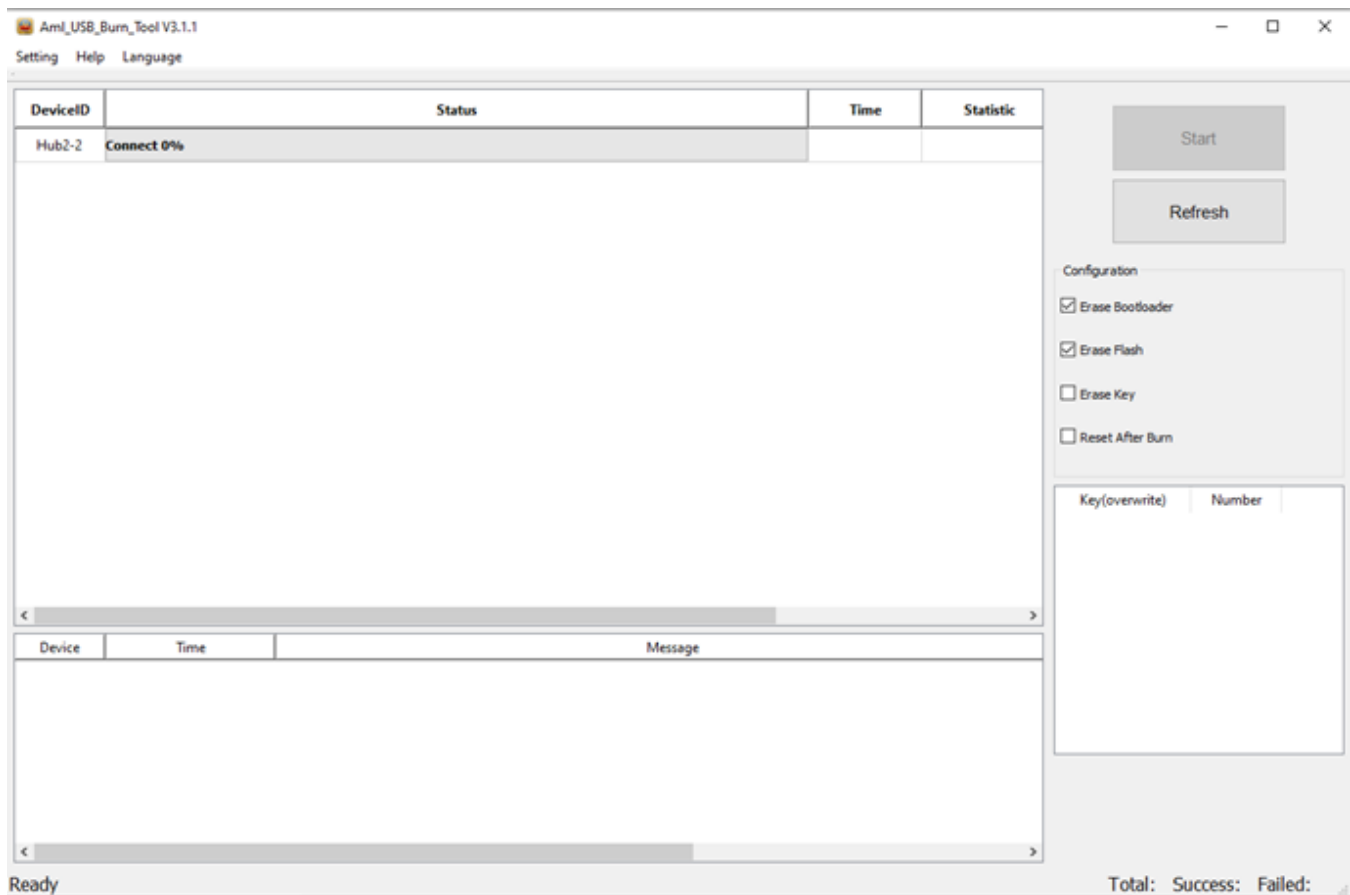
You shall see below bootloader print out on the console window indicating USB burning mode enabled.



Option 2: Apply power and press "enter" on console to access uboot prompt then execute "run update" command to start USB burning mode.



Launch USB Burn Tool V3.1.1 and verify it's connected with AH212. We shall see the "connection 0%" status



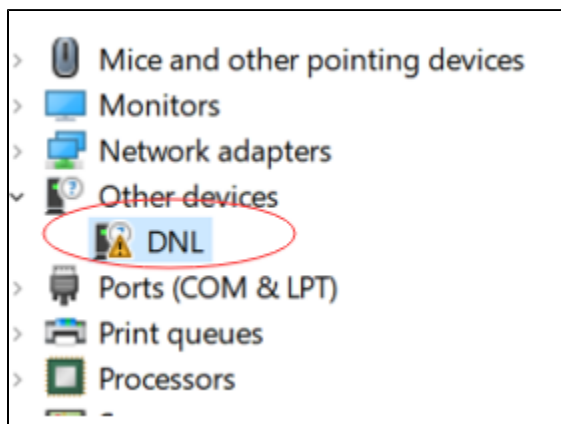
Load the image and start to program the AH212.

Once the flashing is completed, click on STOP and disconnect USB-C cable.

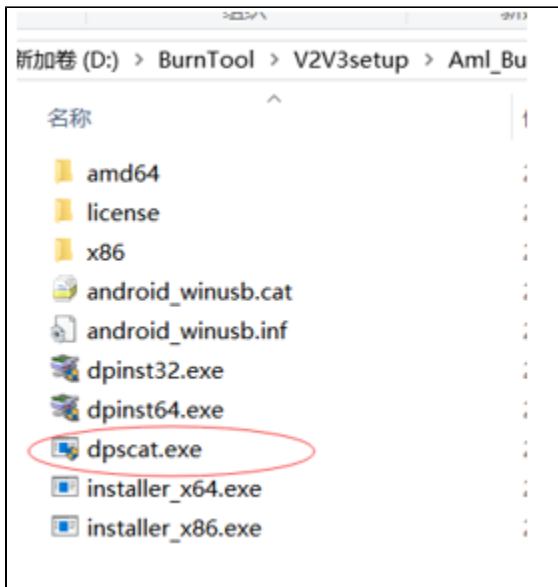
Close the USB Burning Tool App and cycle the AC.

If driver is not properly installed and you cannot see the device , please manually install the driver with below steps:

If a yellow exclamation point "!" appears when the device is connected in the device manager, it means that the driver has not been installed successfully, and you need to install the driver manually.

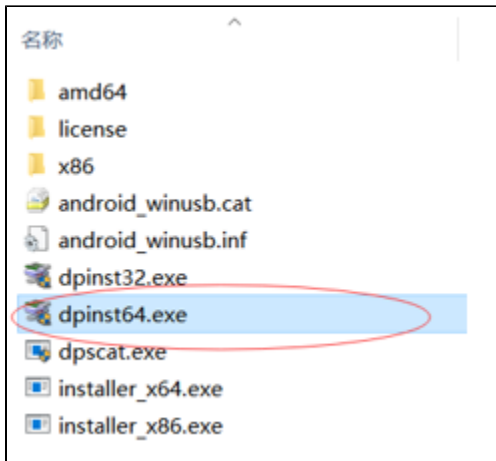


1) First enter the directory specified by the tool installation, first click dpscat.exe to sign the driver.



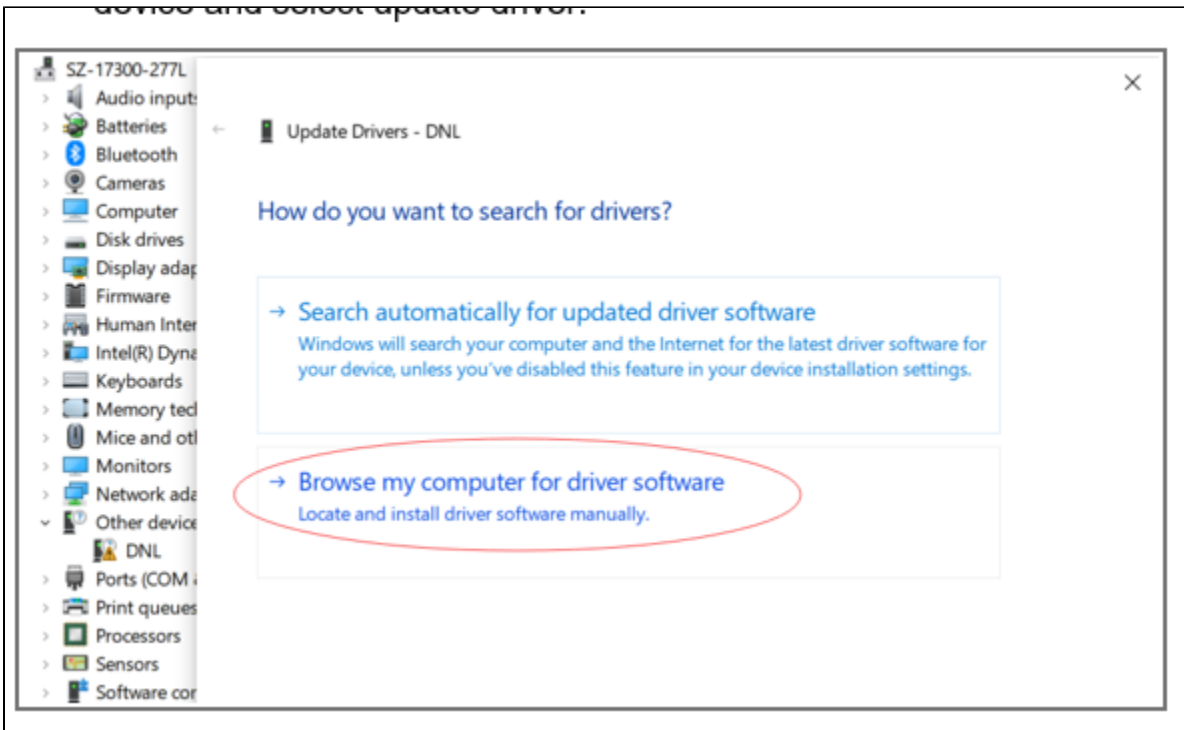
2) During the signing process, if security prompt when need signature or authorization, need allow this operation.

3) Choose the corresponding installation according to the system bit requirement of your own machine, choose dpinst64.exe for 64 bit and dpinst32.exe for 32 bit.

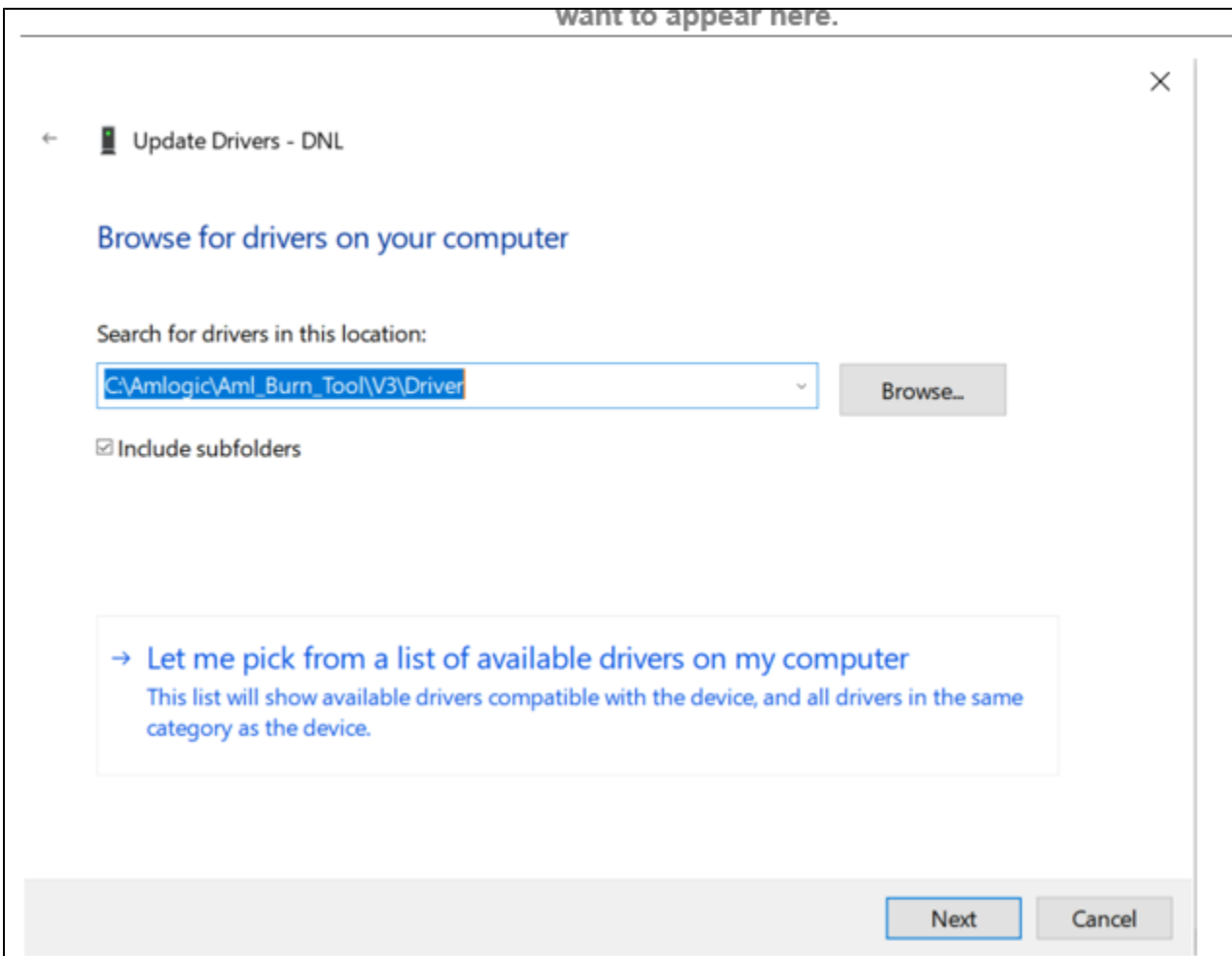


4) Or manually update the driver in the device manager, right-click on the problematic device and select update driver.

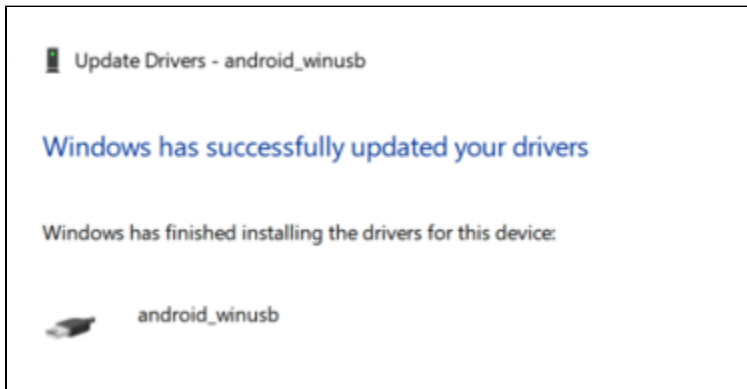
device and select update driver.



5) Select "Browse my computer for driver software", select C:\Amlogic\Aml_Burn_Tool\V3\Driver in the tool installation directory to update the driver.

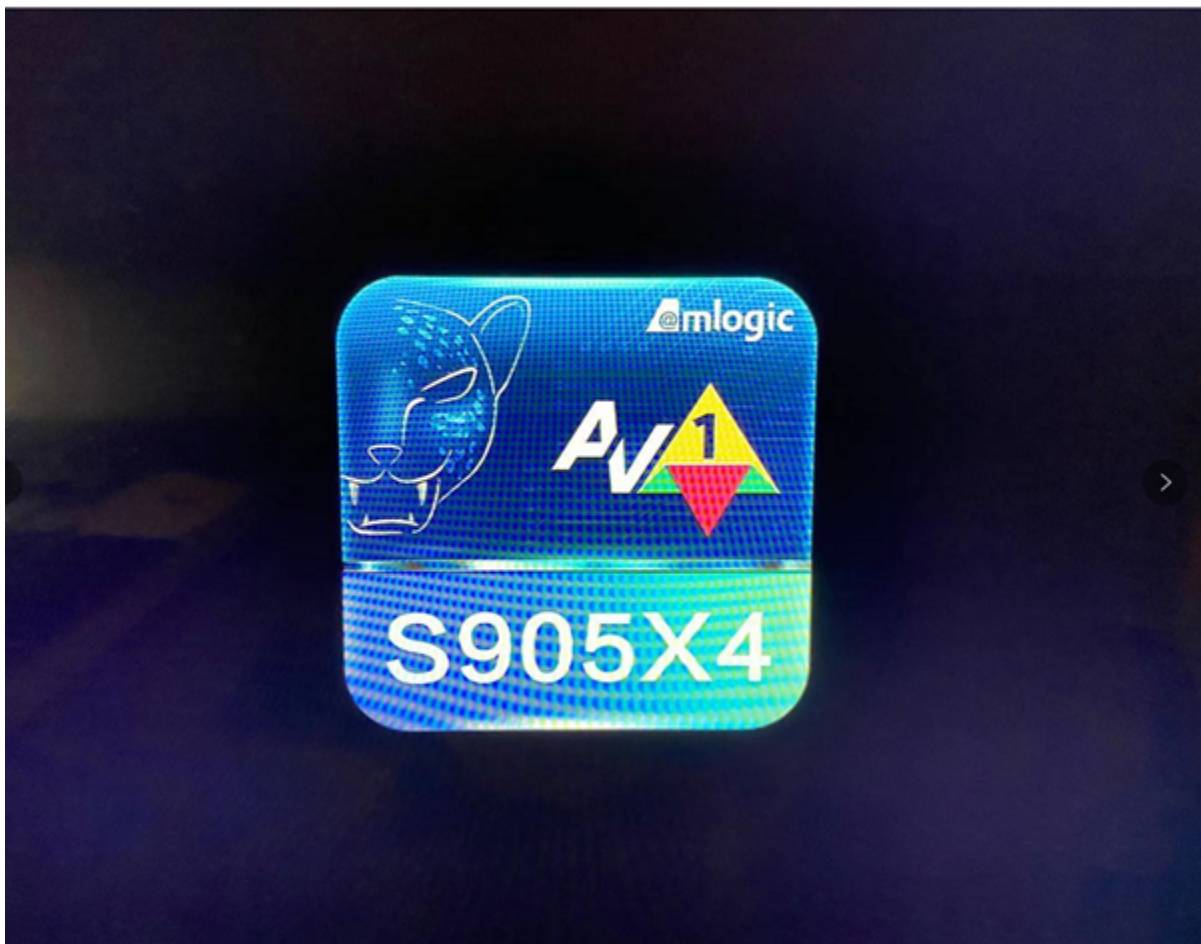


6. The driver is installed successfully.

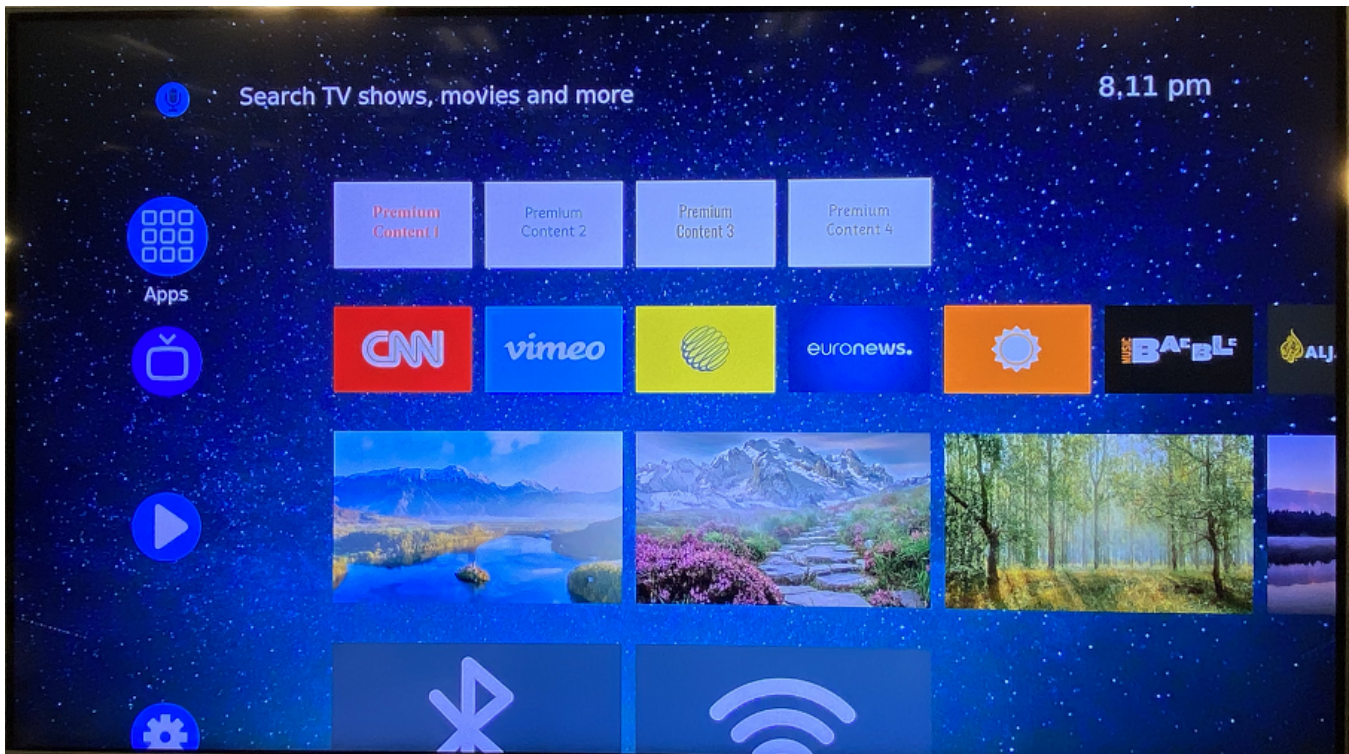


6. Boot up UI

AH212 shall reboot with boot logo as below if the image is programmed successfully and AC cycled.



A generic RDK UI shall be launched with active ethernet.



Use Amlogic IR remote to check YouTube and verify 2K and 4K playback.

Other Methods to update the firmware:

2) MfrUtils

Download the image to /data/ path on the device.

mfrUtil -u /data/aml_upgrade_package.img

Youtube RDK CommonPort Architecture:

