



Arrow Lake S Intel® Platform for Linux* Kernel 6.11 Graphics and Media Driver Integration BKM

Validation Report

WW46, November 2024



You may not use or facilitate the use of this document in connection with any infringement or other legal analysis. You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

All product plans and roadmaps are subject to change without notice.

The products described may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at [intel.com](https://www.intel.com).

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

© Intel Corporation. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries.

*Other names and brands may be claimed as the property of others.

Copyright© 2023-2024 Intel Corporation. All rights reserved.

Contents

1	Overview	5
1.1	Security Considerations.....	5
2	Graphics and Native Media Drivers BKM	6
2.1	Setup	6
2.2	GFX/Native Media FW BKM	6
2.3	Libdrm.....	7
2.4	Mesa	7
2.5	gmmlib	7
2.6	Media-Driver	8
3	Graphics and Native Media Drivers BKM	9
3.1	Verifying HW Decoding.....	9



Revision History

Revision	Description	Revision Date
1.0	• Initial Release	September 2023
2.0	• Updated Kernel Version to 6.5.7	February 2024
3.0	• Updated Kernel Version to 6.8.0	June 2024
4.0	• Updated Kernel Version to 6.9.0	July 2024
5.0	• Updated Kernel Version to 6.10.1	August 2024
6.0	• Updated Kernel Version to 6.11	November 2024

§§

1 Overview

This document covers the BKM for integrating Graphics FW and Media Drivers on **Linux Kernel 6.11**. The document provides steps to integrate the Guc, Huc and DMC Fw and the media driver packages.

1.1 Security Considerations

This document provides a sample integration workflow to enable GFX FW and Media Drivers on Intel® Platform for Linux Kernel 6.11 platform. These instructions shall be used as reference only, and it's the responsibility of the <integrator> to ensure that all components and their dependencies are using the latest available secure versions of the implementation. These instructions are for reference only, hence it's the responsibility of the <integrator> to take security considerations into account prior building a production capable system:-

§§

2 Graphics and Native Media Drivers BKM

2.1 Setup

Install the required packages using the command

- `sudo apt-get install gcc cmake git meson libpciaccess-dev python3 pip llvm nasm libva-dev libva2`
- Please uncomment the “deb-src” lines in `/etc/apt/sources.list`
- `sudo apt update`
- `sudo apt-get build-dep mesa`

2.2 GFX/Native Media FW BKM

Download the below FW from

<https://git.kernel.org/pub/scm/linux/kernel/git/firmware/linux-firmware.git/tree/i915>

and copy it to `/lib/firmware/i915/` on the DUT.

Download the GuC, HuC, GSC FW from below links & copy to `/lib/firmware/i915/` on the DUT

- https://git.kernel.org/pub/scm/linux/kernel/git/firmware/linux-firmware.git/tree/i915/mtl_dmc.bin
- https://git.kernel.org/pub/scm/linux/kernel/git/firmware/linux-firmware.git/tree/i915/mtl_guc_70.bin
- https://git.kernel.org/pub/scm/linux/kernel/git/firmware/linux-firmware.git/tree/i915/mtl_gsc_1.bin
- https://git.kernel.org/pub/scm/linux/kernel/git/firmware/linux-firmware.git/tree/i915/mtl_huc_gsc.bin
- `mtl_dmc.bin` (v2.21)
- `mtl_guc_70.bin` (70.29.2)
- `mtl_huc_gsc.bin` (version 8.5.4)
- `mtl_gsc_1.bin` (cv1.0, r102.0.10.1878, svn 1)

Check that “`i915.enable_guc=3`” is set in the kernel parameters to enable both DMC, GuC, HuC & GSC firmware loading before trying the below steps.

- Run `sudo update-initramfs -u -k all`, in case firmware failed to load.

2.3 Libdrm

Follow the below steps to build and install the libdrm package (libdrm-2.4.120) for the ARL-S platform.

- git clone <https://gitlab.freedesktop.org/mesa/drm.git>
- cd drm
- git checkout libdrm-2.4.120
- meson builddir/
- sudo ninja -C builddir/install

Note: If an error “ninja: fatal: chdir to 'builddir/install' - No such file or directory”, please follow the below steps.

- cd builddir
- ninja
- sudo ninja install

2.4 Mesa

Follow the below steps to build and install the mesa package for the ARL-S platform.

- uncomment the “deb-src” lines in /etc/apt/sources.list.
- sudo apt-get update
- git clone <https://gitlab.freedesktop.org/mesa/mesa.git>
- cd mesa
- git checkout ce1bbd241e
- meson --prefix=/usr/ -Dplatforms="x11,wayland" -Dgallium-drivers="iris" -Dvulkan-drivers=intel -Dvulkan-layers=device-select,overlay build
- ninja -C build/
- sudo ninja -C build/ install

Note: If an error “ninja: error: loading 'build.ninja' : No such file or directory” , please follow the below steps

- cd build
- ninja
- sudo ninja install

2.5 gmmlib

Follow the below steps to build and install the gmmlib package (intel-gmmlib-22.3.18) for the ARL-S platform.

- git clone <https://github.com/intel/gmmlib.git>
- cd gmmlib
- git checkout intel-gmmlib-22.3.18
- Please refer to the steps present at <https://github.com/intel/gmmlib> to build and install the package.

2.6 Media-Driver

Follow the below steps to build and install the Media driver package for the ARL-S platform.

- git clone <https://github.com/intel/media-driver.git>
- cd media-driver
- git checkout intel-media-24.1.0
- Refer to the steps present at <https://github.com/intel/media-driver#intelr-media-driver-for-vaapi> to build and install the package

§§

3 Graphics and Native Media Drivers BKM

3.1 Verifying HW Decoding

Follow the below steps to verify HW decoding on the ARL-S platform.

- `sudo apt-get install mpv vainfo mesa-utils intel-gpu-tools`
- run the below commands:
 - `export LIBVA_DRIVERS_PATH=/usr/lib/x86_64-linux-gnu/dri`
 - `export LIBVA_DRIVER_NAME=iHD`
- Run vainfo to check media driver version. It should show the iHD version as:
vainfo: Driver version: Intel iHD driver for Intel(R) Gen Graphics - 24.1.0 (COMMIT_ID)
- Run `glxinfo | grep Mesa`. The output should show Mesa 24.2.0-devel (COMMIT_ID).
- To use Hardware Decoding with mpv, use the command.
 - `mpv <input.mp4>`
 - Verify with `intel_gpu_top` or `intel_gpu_top -l` to check if VCS/VCS are being utilized.

§§