



## Qualcomm Software Center

80-72780-2 AW

May 8, 2025

# Contents

---

<b>1</b>	<b>Overview</b>	<b>3</b>
<b>2</b>	<b>Features</b>	<b>4</b>
<b>3</b>	<b>Install QSC</b>	<b>5</b>
3.1	Machine requirements . . . . .	5
3.2	Download QSC for Linux . . . . .	5
3.3	Install QSC for Linux . . . . .	7
3.4	Download and install QSC on Windows . . . . .	8
3.5	Download and install QSC on Windows Subsystem for Linux (WSL) . . . . .	9
3.6	Download and install QSC on UTM for Mac . . . . .	13
<b>4</b>	<b>Launcher</b>	<b>19</b>
4.1	Use the QSC Launcher GUI . . . . .	19
<b>5</b>	<b>QSC CLI</b>	<b>26</b>
5.1	Use the QSC CLI . . . . .	26
<b>6</b>	<b>Tools and SDKs</b>	<b>30</b>
6.1	Find a tool or SDK . . . . .	30
6.2	Install a tool or SDK . . . . .	31
6.3	Manage licenses . . . . .	34
6.4	Use the QSC CLI . . . . .	34
6.5	Deprecated qpm-cli commands . . . . .	36
<b>7</b>	<b>Chip software</b>	<b>38</b>
7.1	Download software product . . . . .	38
7.2	Download software image . . . . .	40
<b>8</b>	<b>Troubleshooting</b>	<b>42</b>
8.1	Log in issues . . . . .	42
8.2	Firewall requirements . . . . .	42
8.3	Contact support . . . . .	47

# 1 Overview

---

The Qualcomm® Software Center (QSC) manages Qualcomm software and simplifies the development process.

- Browse Qualcomm tools, SDKs, and chip software.
- Manage download, installations, and updates for tools, SDKs, and chip software.
- Download, compile, manage, and load software to connected devices (referred to as **Launcher** on QSC).

## 2 Features

---

The QSC ecosystem offers the following features:

- Browse and discover Qualcomm tools, SDKs, and chip software in the web [portal](#) or the Qualcomm Software Center GUI desktop application.
- The desktop application is a guided UI to find, download, compile, and manage software resources for Qualcomm devices (referred to as “[Launcher](#)” on QSC). You can also manage installation and updates of tools, SDKs, and chip software for your development environment.
- The `qsc-cli` command line interface (CLI) downloads and compiles Qualcomm software builds. [qsc-cli](#) uses the Docker environment at the back end and simplifies host set up, sync, and build to a few simple commands. [qsc-cli](#) also manages tool and SDK download and installation.

# 3 Install QSC

---

To get started, register with [Qualcomm](#) to get a Qualcomm ID.

## 3.1 Machine requirements

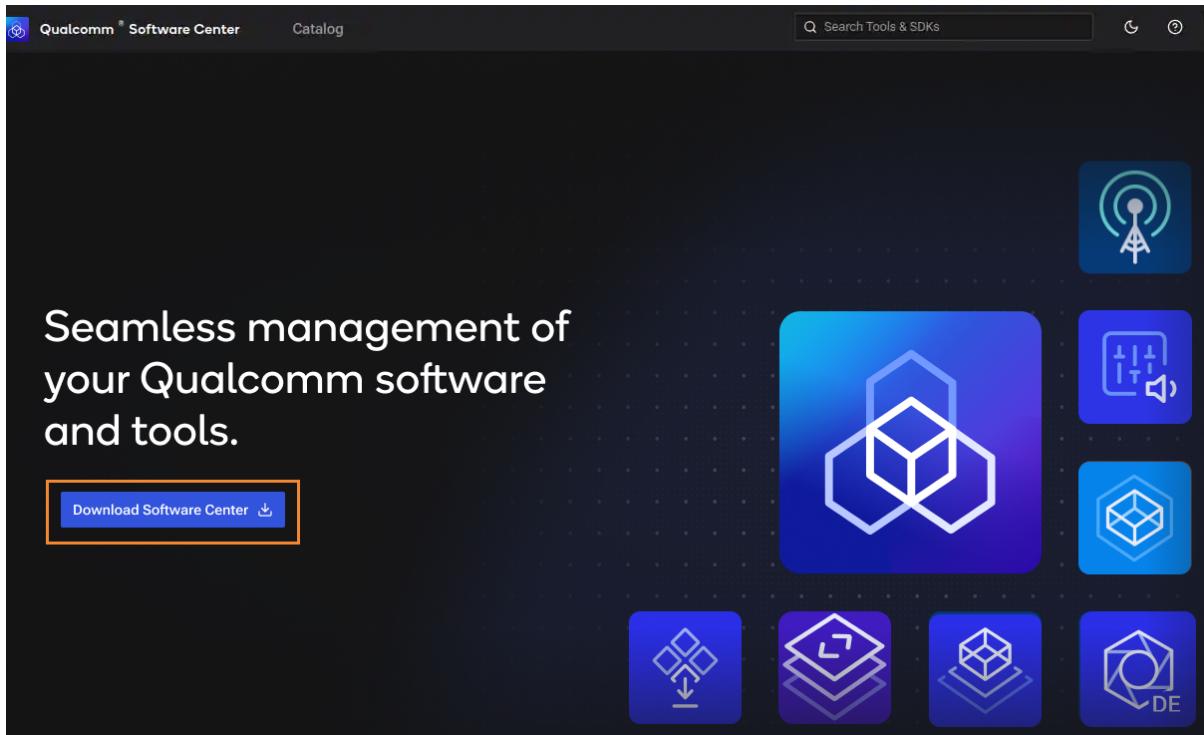
- Linux (Ubuntu 22.04) or Windows. The Launcher workflow is supported on Linux and Windows (WSL).
- Minimum available storage of 300 GB to download and compile software product builds.
- Minimum of 16 GB RAM.
- Internet connectivity to Qualcomm back end servers (see [firewall requirements](#)).

## 3.2 Download QSC for Linux

Download Qualcomm Software Center for Linux using the following options.

## GUI

1. Use your Qualcomm ID to log in to <https://softwarecenter.qualcomm.com>, and click on **Download Software Center**.



This takes users to the Qualcomm Software Center tools detail page.

2. Select the preferred OS configuration and click **Download**.

The QSC Debian package downloads to your machine.

## CLI

1. If not already installed, run the following command to install curl.

```
sudo apt install curl
```

2. To download the QSC installer for x86, run the following command.

```
curl -L https://softwarecenter.qualcomm.com/api/download/
software/tools/Qualcomm_Software_Center/Linux/Debian/latest.deb
-o qsc_installer.deb
```

---

**Note:** For ARM64, the installer path is <https://softwarecenter.qualcomm.com/api/download/software/qsc/linux-arm64/latest.deb>.

---

## 3.3 Install QSC for Linux

Install Qualcomm Software Center for Linux using the following options.

### CLI

To install the Debian package, run the following command.

```
sudo apt update
```

```
sudo apt install ./qsc_installer.deb
```

---

**Note:** Ignore the following error message if it appears:

```
o/p:: N: Download is performed unsandboxed as root as file
'/home/ubuntu22/qsc_installer.deb' couldn't be accessed by user '_'
apt'. - pkgAcquire::Run (13: Permission denied)
```

---

After a successful install, a Installed Qualcomm Software Center v1.X.X successfully message displays.

---

**Note:** For the Launcher workflow to detect connected devices and flash software builds, PCAT and QUD must be installed on the host machine. Follow the instructions to use the [qsc-cli](#) to install PCAT and QUD.

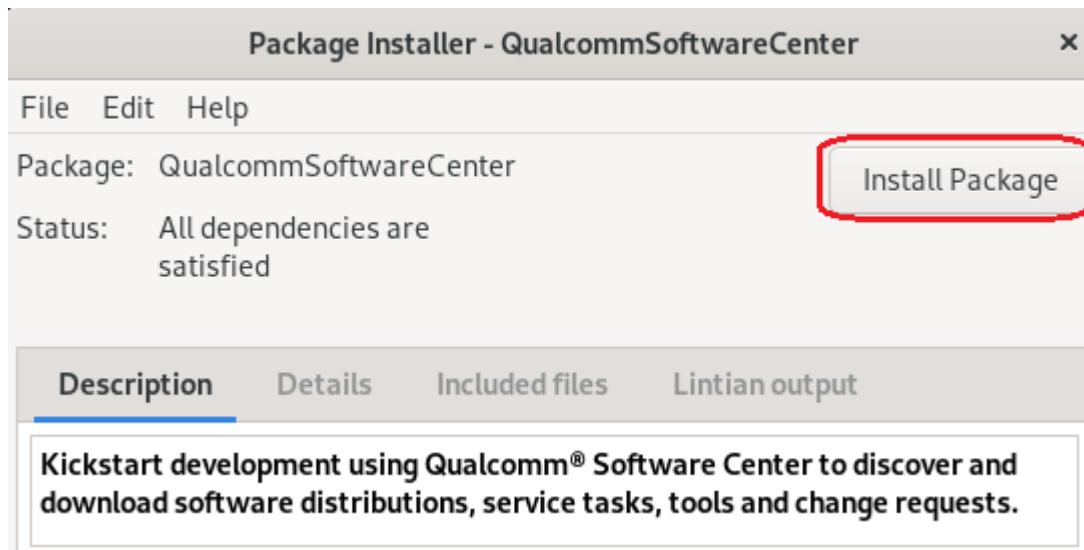
---

## GUI

Install the downloaded QSC Debian package using the Ubuntu GNOME GUI (GDebi).

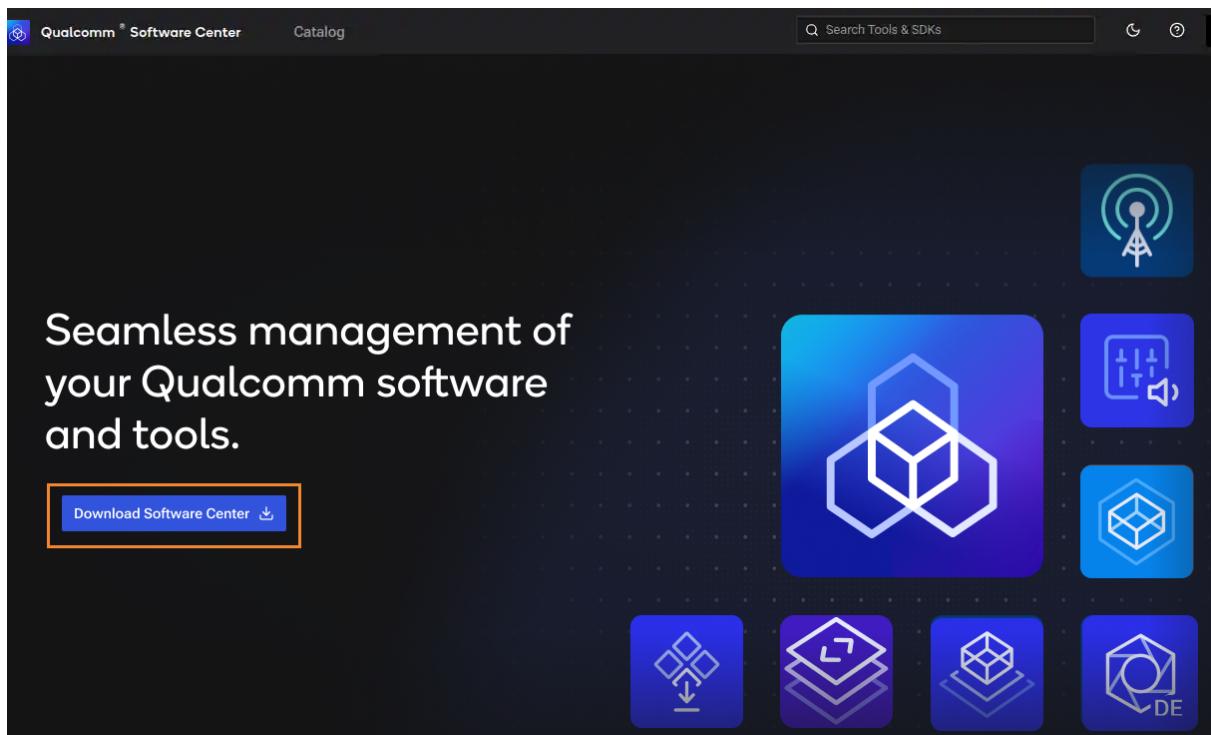
To install GDebi, run the following command in a terminal window.

```
sudo apt install gdebi
```



## 3.4 Download and install QSC on Windows

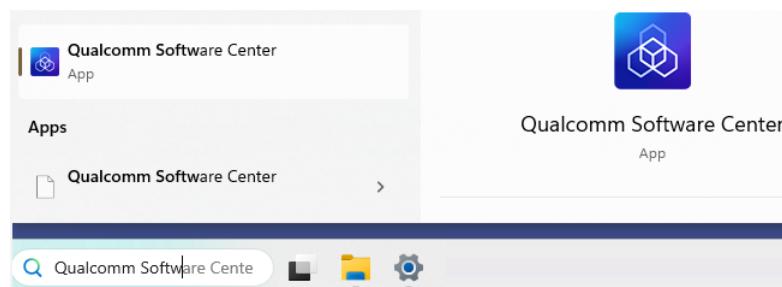
1. Log in to <https://softwarecenter.qualcomm.com/> with your Qualcomm ID.
2. Click on **Download Software Center**.



The QSC executable downloads to your Windows machine.

3. Click the QSC .exe executable to start the install.

To open the Qualcomm Software Center desktop application after installation, search for **Qualcomm Software Center** in the Windows Search box.



### 3.5 Download and install QSC on Windows Subsystem for Linux (WSL)

Use WSL to download and install QSC.

For additional troubleshooting, refer to [WSL support](#).

## Machine requirements

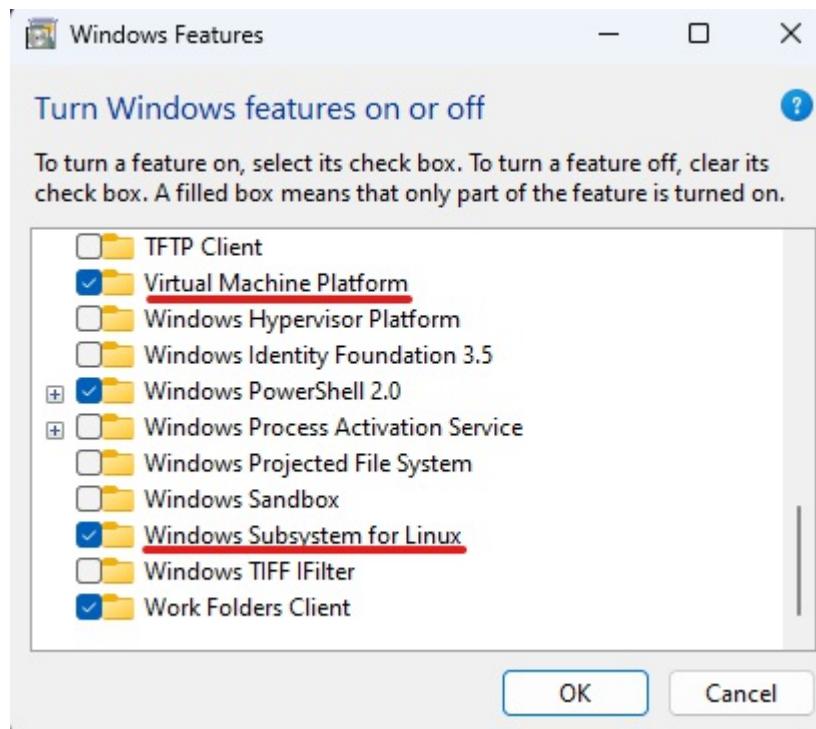
- Windows 11 or Windows 10 Build 18362.1049 or later.
- Minimum of 32 GB RAM. Follow the instructions at [Speed up qsc-cli operations](#) to allocate available RAM to WSL.
- Minimum available storage of 500 GB.

## Prerequisites

Before installing WSL2, ensure the following Windows features are enabled.

- Microsoft-Windows-Subsystem-Linux
- VirtualMachinePlatform

To enable them, search for **Turn Windows features on or off** in the Windows search bar. A new window will pop up with a list of features available. Please ensure that the Microsoft-Windows-Subsystem-Linux and VirtualMachinePlatform features are enabled as shown below:



**Note:** If you had to enable any or both of the above features, Windows may ask you to restart the machine. Please restart your machine first and then proceed with the next steps.

## Install WSL and QSC

1. Install WSL using the procedure at <https://learn.microsoft.com/en-us/windows/wsl/install>.

To set WSL2 as the default WSL version, run the following command in Windows PowerShell in admin mode.

```
wsl --set-default-version 2
```

2. Install Ubuntu distro by running the following command in Windows PowerShell in admin mode.

```
wsl --install -d Ubuntu-22.04
```

Once the installation completes, create a new username and password. You will be logged in by default as that user upon account creation.

3. Install qsc-cli by running the following commands in the Ubuntu session.

```
curl -L https://softwarecenter.qualcomm.com/api/download/software/tools/Qualcomm_Software_Center/Linux/Debian/latest.deb -o qsc_installer.deb
```

```
sudo apt update
```

```
sudo apt install ./qsc_installer.deb
```

---

**Note:** Ignore the following error message if it appears:

```
o/p:: N: Download is performed unsandboxed as root as file '/home/ubuntu22/qsc_installer.deb' couldn't be accessed by user '_apt'. - pkgAcquire::Run (13: Permission denied)
```

---

4. Wait ~5 minutes and install the docker with the following commands.

```
curl -fsSL https://get.docker.com -o get-docker.sh
```

```
sudo sh get-docker.sh
```

```
sudo usermod -aG docker $USER
```

```
sudo service docker start
```

5. Set the docker to start automatically by opening /etc/wsl.conf and adding the following line (if not present).

```
[boot]
command = sudo service docker start
```

6. On a Windows PowerShell in admin mode, run the following command for the changes to take effect.

```
wsl --shutdown
```

---

**Note:** For the Launcher workflow to detect connected devices and flash software builds, PCAT and QUD must be installed on the Windows host machine. Follow the instructions to use the [qsc-cli](#) to install PCAT and QUD.

---

## Postrequisites

After installing WSL, the following optimizations are required.

### Optimize WSL performance

By default, Ubuntu installs on your C drive, which may not have enough space. If you have another drive with more storage space, you can change your Ubuntu distro's install location to optimize performance.

1. Run the following command in Windows PowerShell in admin mode.

```
wsl --shutdown
```

2. From the Windows search bar, open **Add or Remove Programs**.
3. Locate Ubuntu-22.04 in the list of apps.
4. Click the app and then click **Move**.
5. Use the dropdown to select a new location and click **Move**.

---

**Note:** It may take some time for the move to complete.

---

### Optimize QSC CLI operation speed

The system limits RAM usage by WSL to 50% of the total RAM on Windows. For example, even if you have 32 GBs of RAM available on the host, only 16 GBs would be allocated to WSL. A workaround for this limitation is to update the .wslconfig file where you can allocate available RAM to WSL.

1. Create a %UserProfile%.wslconfig file with the following content.

---

**Note:** These values are just for reference. Please crosscheck each of these settings to match the specifications on your machine.

---

```
# Settings apply across all Linux distros running on WSL 2
[wsl2]
# Limits VM memory to use no more than 64 GB, this can be set as
whole numbers using GB or MB
memory=64GB
# Sets the VM to use 16 virtual processors.
processors=16
# Sets amount of swap storage space to 64GB, default is 25% of
available RAM
swap=64GB
# Sets swapfile path location, default is %USERPROFILE%\AppData\Local\Temp\swap.vhdx
swapfile=E:\\wsl-swap.vhdx
# Enable experimental features
[experimental]
# Automatically releases cached memory after detecting idle CPU
usage. Set to gradual for slow release, and dropcache for
instant release of cached memory.
autoMemoryReclaim=dropcache
```

2. Run the following command in Windows PowerShell in admin mode.

```
wsl --shutdown
```

3. Launch the Ubuntu distro again to start using [qsc-cli](#).

---

**Note:** Only [qsc-cli](#) is supported on WSL. The Qualcomm Software Center desktop application may not work on WSL.

---

## Support

- For common WSL related issues, refer to [Microsoft troubleshooting](#).
- For network related issues, refer to [Microsoft GitHub](#).

## 3.6 Download and install QSC on UTM for Mac

Use UTM for Mac to download and install QSC.

## Machine requirements

- Host operating system: macOS Big Sur 11.3 or later
- Architecture: ARM64
- Processors: ARM-based Apple Silicon processor
- Minimum of 18 GB RAM
- Minimum available storage of 500 GB (300 GB free space)

## Download UTM and set up VM

1. Download the applicable Ubuntu ISO image from the Ubuntu website.

If your Mac laptop uses ARM architecture, download the ARM ISO image.

1. Go to <https://ubuntu.com/download/server/arm>.
2. Click **Alternative and previous releases**.
3. Download the 22.04 ISO image.
2. Download and install UTM.
  1. Go to <https://mac.getutm.app/> and click **Download**.
  2. Double-click the downloaded .dmg file and drag and drop UTM into you Applications folder.
  3. Go to Applications and open UTM.
3. Install Ubuntu.
  1. Click **Create a New Virtual Machine**.
  2. Click **Virtualize**.
  3. Select **Linux** as the operating system. For **Boot ISO Image**, click **Browse** to navigate to and select the previously downloaded Ubuntu ISO image file.
  4. Click **Continue**
  5. Select RAM and CPU cores (the minimum recommended RAM and cores are 10 GB and 6 cores, respectively) and click **Continue**. For better performance use 70% of the host configuration.

---

**Note:** To check the system configuration on your Mac, click Apple menu > **System Settings**, then click **General** in the sidebar. You may need to scroll down. On the right, click **About**, then click **System Report**.

---

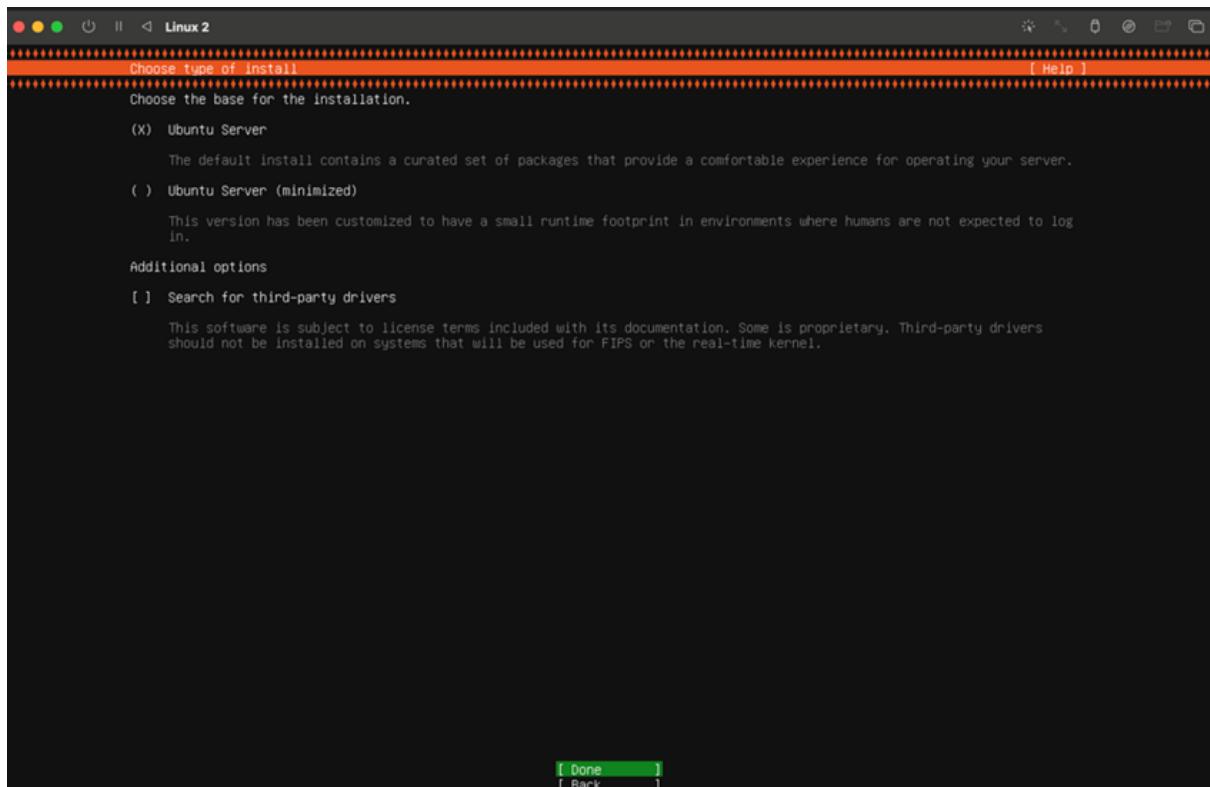
6. Specify the storage size (the minimum recommended storage is 300 GB to download and compile software product builds) and click **Continue**.
  7. Select the shared directory that can be accessible inside the VM (for example, the download folder) and click **Continue**.
  8. Update the system name (default is *Linux*) and click **Save** to save the system configurations.
4. Open and set up Ubuntu by clicking the virtual machine play button.

---

**Note:** The Tab key will move the cursor. The Space key will select the option.

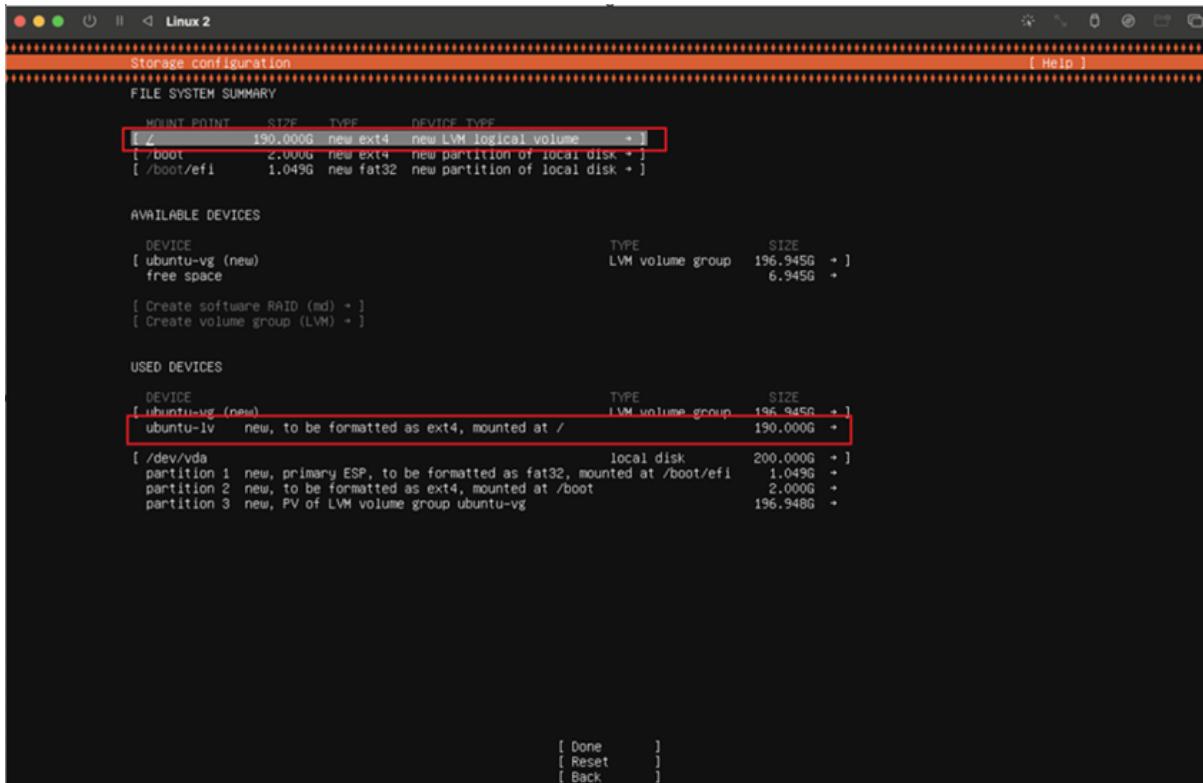
---

1. Select **Try or Install Ubuntu Server**.
2. Select the preferred language.
3. Select the keyboard layout.
4. Select **Ubuntu Server** (default) as the installation type.



5. Select the default network interface.

6. Select the default Ubuntu archive mirror.
7. Select the default guided storage configuration.



8. On the Storage Configuration page, go to **USED DEVICES**, press tab to select **ubuntu-lv new, to be formatted as ext4, mounted at** and press enter to change the **ubuntu-lv size** (for example, if **ubuntu-vg** has 196, keep **ubuntu-lv** as 190).
9. On the Confirm destructive action page, select **Continue**.

---

**Note:** This does not erase content inside the macOS.

---

10. On the Profile setup page, input the necessary information for the Ubuntu VM.

---

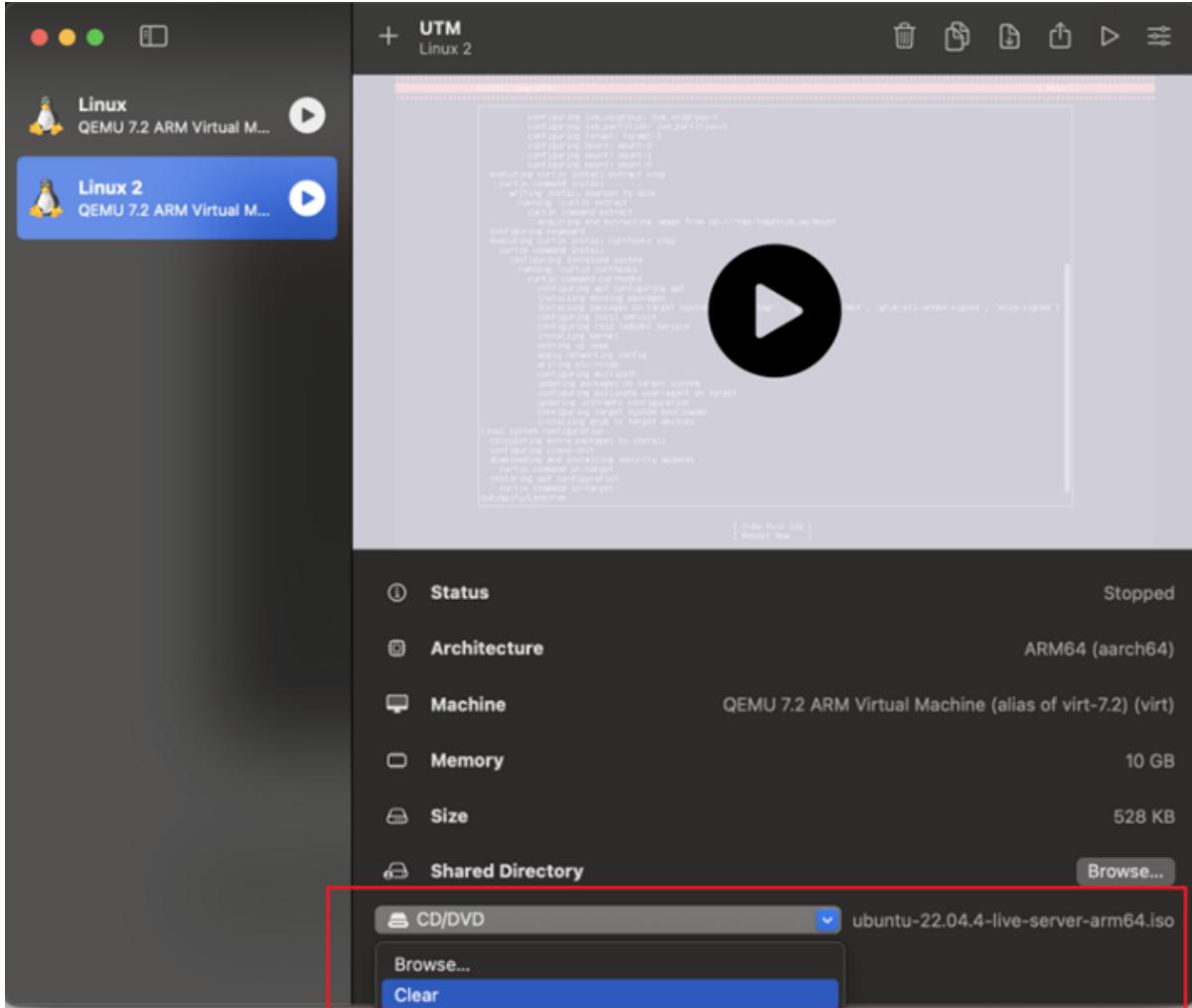
**Note:** The username and password will become the login credentials for the VM.

---

11. On the Upgrade to Ubuntu Pro page, select **Skip for now**.
12. If you want to enable SSH for your server, install the OpenSSH server package. To skip

SSH setup, select **Done**.

- The Ubuntu system will install. Once complete, dismount the CD/DVD from the VM and restart.



- To use the VM, click the play button and log in. It may take time to show the Ubuntu login screen. If it fails to appear, restart UTM and start the VM.
- Install Ubuntu desktop.
  - In the Ubuntu terminal, run the command to update your system.
 

```
sudo apt update && sudo apt upgrade -y
```
  - Install Ubuntu desktop.

```
sudo apt install ubuntu-desktop -y
```

3. Reboot the system.

```
reboot
```

Ubuntu desktop is now installed.

7. Connect a USB to the machine. The VM will ask to allow the USB to connect.

On Ubuntu desktop, click the USB symbol in the top right corner to view recently connected USBs. A list of available USBs can be accessed in the Ubuntu terminal by running the command.

```
lsusb
```

8. After setup is complete, install qsc-cli by running the following commands in the Ubuntu session.

```
curl -L https://softwarecenter.qualcomm.com/api/download/software/qsc/linux-arm64/latest.deb -o qsc_installer.deb
```

```
sudo apt update
```

```
sudo apt install ./qsc_installer.deb
```

---

**Note:** Ignore the following error message if it appears:

```
o/p:: N: Download is performed unsandboxed as root as file  
'/home/ubuntu22/qsc_installer.deb' couldn't be accessed by user  
'_apt'. - pkgAcquire::Run (13: Permission denied)
```

---

Start using [qsc-cli](#) to download and compile software product builds.

---

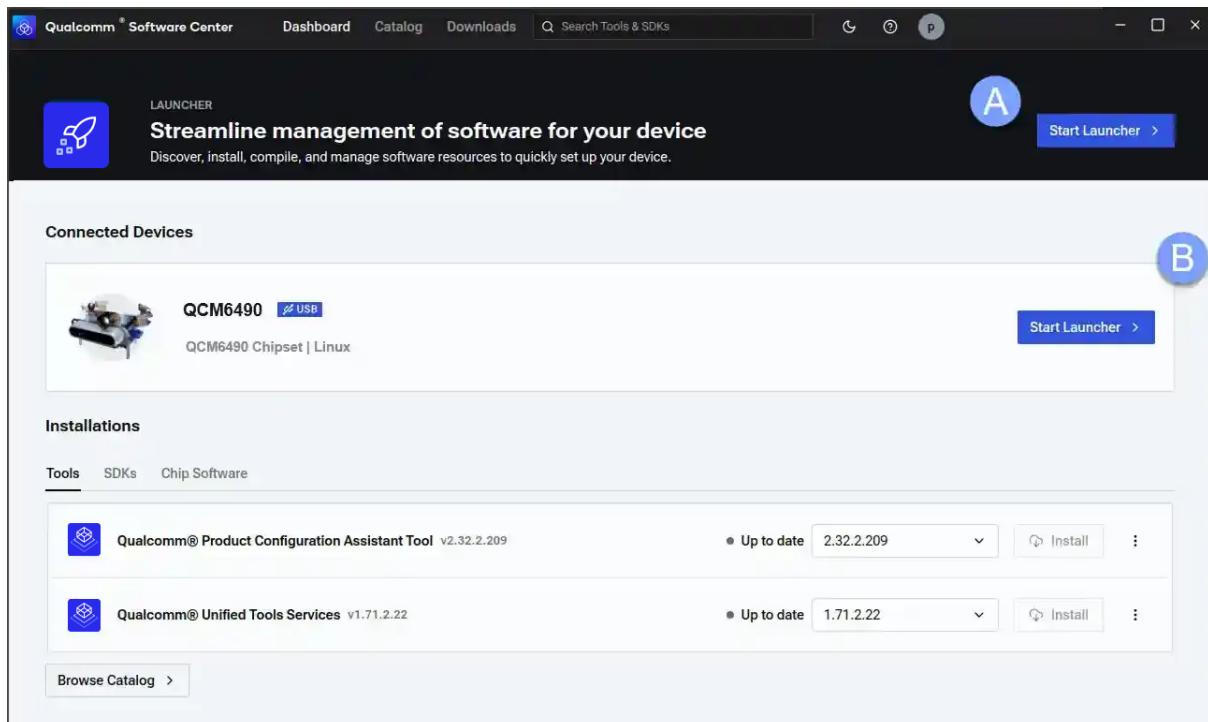
**Note:** For the Launcher workflow to detect connected devices and flash software builds, PCAT and QUD must be installed on UTM. Follow the instructions to use the [qsc-cli to install PCAT and QUD](#).

---

# 4 Launcher

## 4.1 Use the QSC Launcher GUI

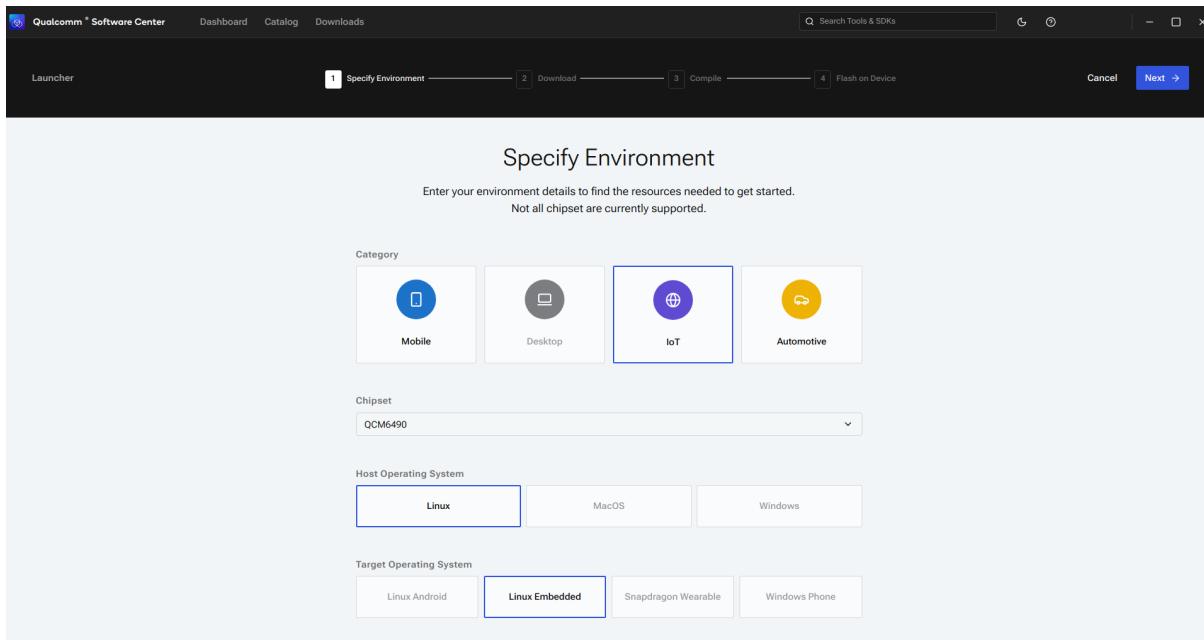
1. To open the Launcher desktop application, search for **Qualcomm Software Center** in the **Applications** menu.
2. Use your Qualcomm ID to log in to the Qualcomm Software Center desktop application. A dashboard similar to the following image displays.



- If you do not have a connected device, click **Start Launcher** (A) on the top panel to start the steps to configure, download, compile, and load software to set up your device.
- If you have a connected device, click **Start Launcher** (B) for the appropriate device in

the **Connected devices** panel to start the download, compile, and load steps to set up your connected device.

3. On the **Specify Environment** page, select the appropriate values for **Category**, **Chipset**, **Host Operating System** and **Target Operating System**.



4. Click **Next**.

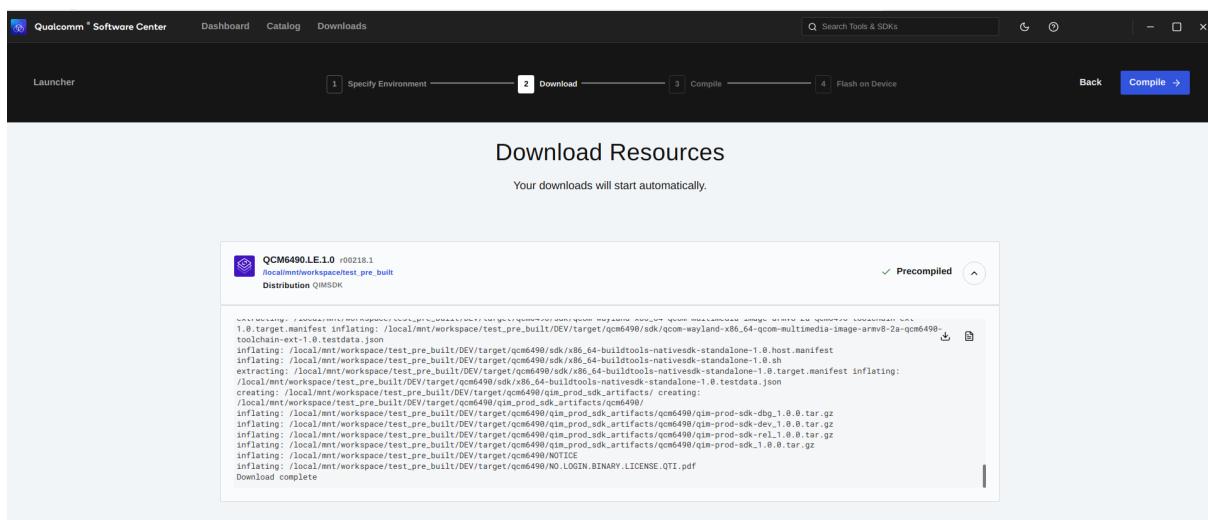
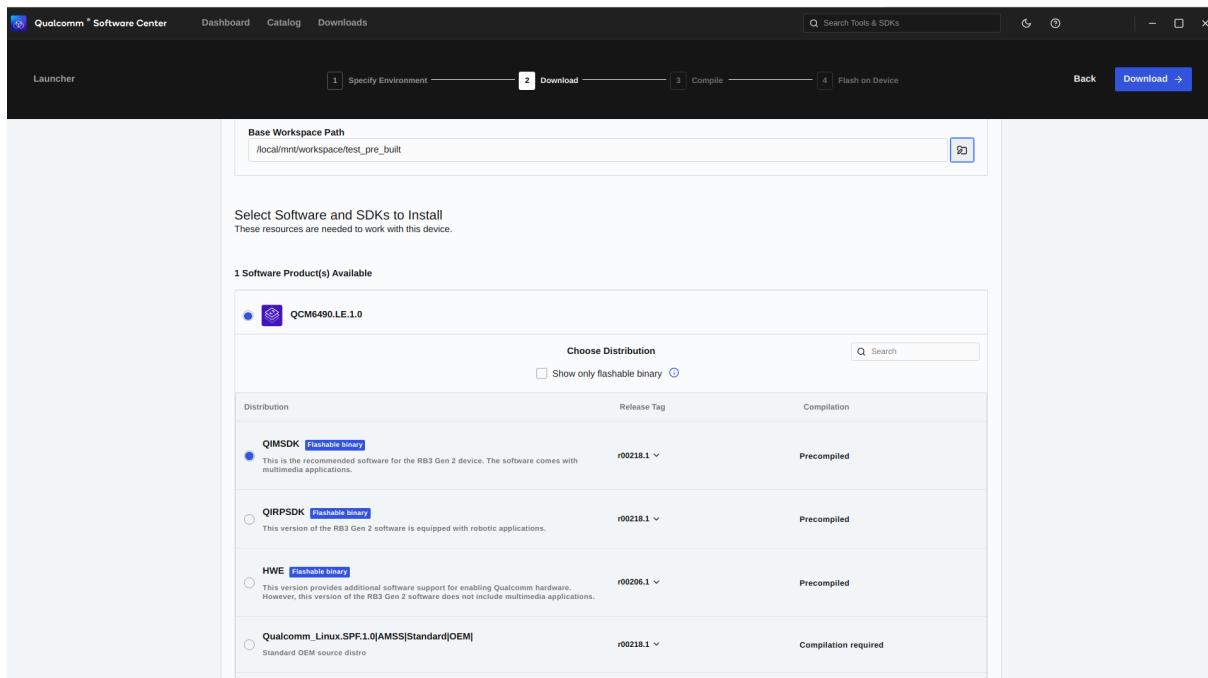
The **Select Resources** page displays.

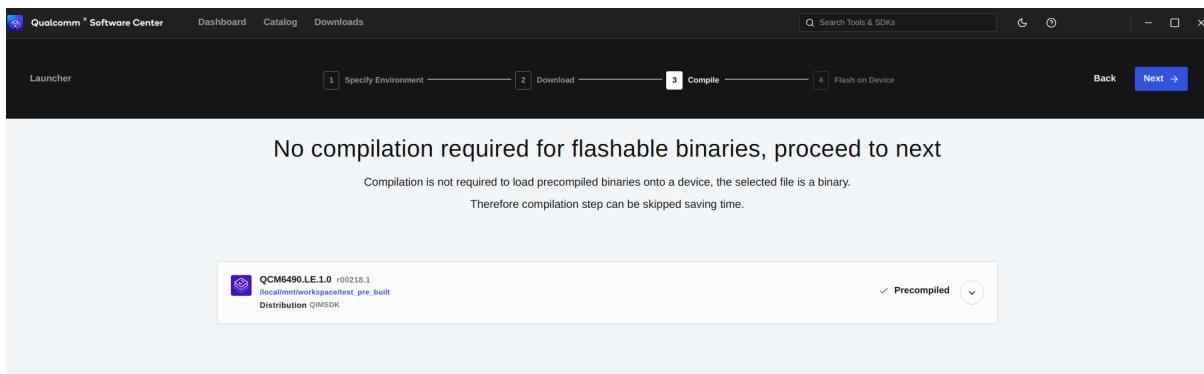
5. On the Select Resource page, do the following:

- In the **Base Workspace Path** field, specify an absolute path to the directory on the host machine to download the software. To display the directory selection window, click the select icon.
- Select the **Software product**.
- Select a distribution and release tag.

6. For prebuilt binaries, select a precompiled flashable binary and click **Download**.

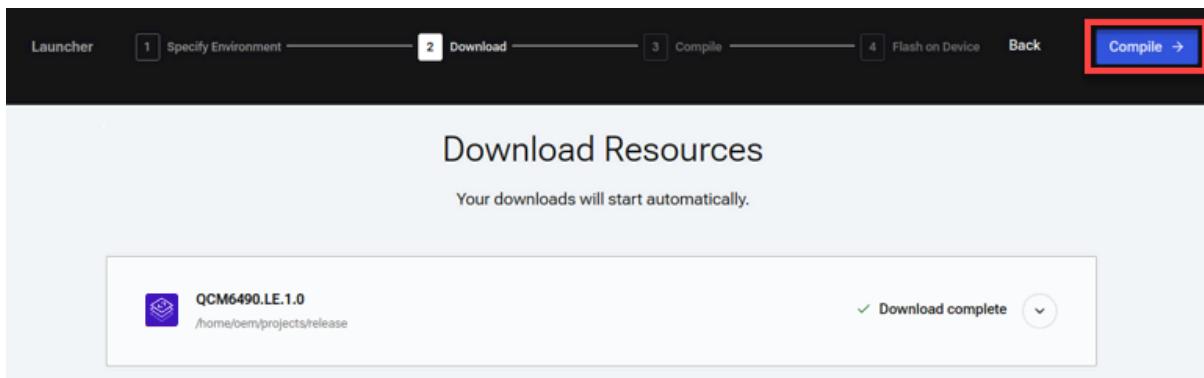
Prebuilt binaries do not require compilation. If a prebuilt binary is selected, follow the onscreen prompts to flash to a connected device.





#### 7. For compilable distributions:

1. Select the compilable distribution and click **Download**.
2. Wait for the download to complete and select **Compile**. Depending on size of the software and host machine configuration, compilation can take a few hours.



3. To view the compilation progress of individual software images, expand the logs panel (as the following image shows).

The screenshot shows the 'Compile Software' page. At the top, there is a navigation bar with steps: 'Specify Environment' (step 1), 'Download' (step 2), 'Compile' (step 3, highlighted in blue), and 'Flash on Device' (step 4). Below the navigation bar, the title 'Compile Software' is displayed. A message states: 'Compilation is required to load software onto a device, and may take a long time. You can navigate away from this page and view progress on the [Downloads](#) page at any time.' The main area shows a log for 'QCM6490.LE.1.0' under 'LE.QCLINUX.1.0.r1'. The log indicates the process is 'Compiling' and shows a progress bar. A 'Logs' dropdown menu is open, showing 'Compiling complete' and a 'Logs' section with a log entry:

```

QCM6490.LE.1.0
> 2024-03-12T18:12:23: 925439de14fa: Verifying Checksum
> 2024-03-12T18:12:23: 925439de14fa: Download complete
> 2024-03-12T18:12:23: 725be5fe0ee1: Verifying Checksum
> 2024-03-12T18:12:23: 725be5fe0ee1: Download complete
> 2024-03-12T18:12:42: 22f08b678a4f: Download complete
> 2024-03-12T18:12:49: Pull complete
> 2024-03-12T18:12:49: Pull complete
> 2024-03-12T18:12:49: Digest: sha256:bb2ff0505779c51084e445b75cff0d5fac20ec8a765bc168933350a380236cd8
> 2024-03-12T18:12:49: Status: Downloaded newer image for 032693710308.dkr.ecr.us-west-2.amazonaws.com/stormchaser/le.um-
k2c:20.04.20231226052639142.8
> 2024-03-12T18:12:50: Starting container with UID : 207156
> 2024-03-12T18:12:51: Finished running build successfully

```

**Tip:** BitBake fetch errors are typically intermittent fetch failures. Retry the [compile step](#) to get past these intermittent errors.

- To flash on device, select the device on which to flash compiled software from the list of connected devices, and select a build flavor if applicable. Be sure to select the correct target device when multiple devices are connected to the host machine.

The screenshot shows the 'Select Device' page. At the top, there is a navigation bar with steps: 'Specify Environment' (step 1), 'Download' (step 2), 'Compile' (step 3), and 'Flash on Device' (step 4, highlighted in blue). Below the navigation bar, the title 'Select Device' is displayed. A message says: 'Choose a device on which to load software.' A list of connected devices is shown in a table:

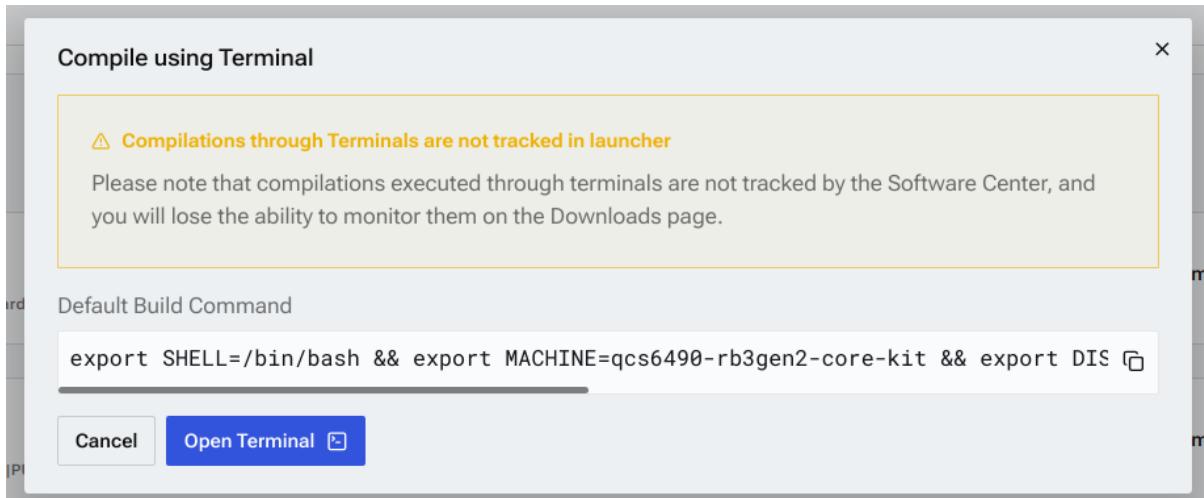
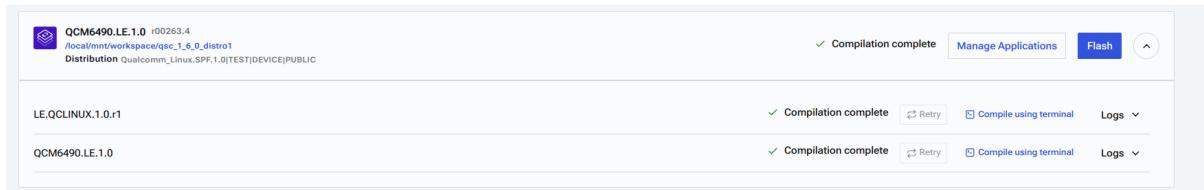
Device	Description
QCM6490	Qualcomm USB Composite Device:QCM6490_bdfffc1a QCS6490 Linux

- For optional advanced customization, execute build commands for specific software image(s) by clicking **Compile using terminal** next to the software image(s). A GNOME terminal will open.

---

**Note:** Compilations executed through the terminal are not tracked by Qualcomm Software Center and the ability to monitor their status on the Download page will be lost.

---



```
File Edit View Search Terminal Help
You can now run 'bitbake <target>'

### Shell environment set up for builds. ###

You can now run 'bitbake <target>'

Common targets are:
  core-image-minimal
  core-image-full-cmdline
  core-image-sato
  core-image-weston
  meta-toolchain
  meta-ide-support

You can also run generated qemu images with a command like 'runqemu qemux86'

Other commonly useful commands are:
 - 'devtool' and 'recipetool' handle common recipe tasks
 - 'bitbake-layers' handles common layer tasks
 - 'oe-pkgdata-util' handles common target package tasks
Loading cache: 100% [########################################] Time: 0:00:02
Loaded 9654 entries from dependency cache.
```

# 5 QSC CLI

---

---

**Note:** These commands are applicable for version 1.13.x onwards.

---

## 5.1 Use the QSC CLI

Use the qsc-cli command-line interface (the GUI version is referred to as [Launcher](#) in the Qualcomm Software Center desktop application) to download, compile, and flash Qualcomm software to connected devices.

For qsc-cli usage on WSL, start a new Ubuntu session via the Windows PowerShell. Avoid using `/mnt` for workspace creation, as it may require admin mode and compilation will be slower.

### Help

Display the qsc-cli help menu for options and updates.

```
qsc-cli chip-software -h  
qsc-cli chip-software download -h  
qsc-cli chip-software compile -h
```

## Log in

Log in with a Qualcomm ID.

```
qsc-cli login -u <your username>
```

List personal access tokens (PATs).

```
qsc-cli show-access-token
```

List user information such as logged-in user and customer ID.

```
qsc-cli show-user-profile
```

## Manage workspaces

List a workspace.

```
qsc-cli chip-software list-workspace
```

Delete a workspace.

```
qsc-cli chip-software delete-workspace --workspace-path <workspace path>
```

## List products, distributions, and releases

List products, distributions, and releases that are available to download and compile. Run the following command to get more information.

```
qsc-cli chip-software info --help
```

## Download

To download both proprietary and open source software from different sources, specify the following input parameters:

- --workspace-path <workspace path>
- --product <Product\_ID>
- --distribution <Distro>
- --release <release>

For example:

```
qsc-cli chip-software download --workspace-path <workspace path> --  
product "QCM6490.LE.1.0" --distribution "Qualcomm_Linux.SPF.1.0|AMSS|  
Standard|OEM|" --release 'r00155.4'
```

## Compile

To compile a downloaded software product, specify an input workspace path.

```
qsc-cli chip-software compile --workspace-path <workspace path>
```

Compile a software image.

```
qsc-cli chip-software compile --workspace-path <workspace path> --  
image "LE.QCLINUX.1.0.R1"
```

## Flash

Flash a device. The --buildflavor argument is optional and only required for devices that have multiple flavors.

```
qsc-cli chip-software flash --workspace-path <workspace path> --  
buildflavor "sa2150p_emmc" --serialnumber "1424354"
```

List build flavors.

```
qsc-cli chip-software flash --workspace-path <workspace path> --list-  
buildflavor
```

List attached devices and get serial number.

```
qsc-cli chip-software device --list
```

## Open build environment

For optional advanced customization, open a build environment for a workspace and execute build commands for the input software image.

```
qsc-cli chip-software open-build-env --workspace-path <workspace path> --image <software image>
```

---

**Note:** This will set up an environment for you to execute your own build commands for a given software image. QSC will not track the status of input workspaces going forward and flash via qsc-cli will not be supported for this workspace.

---

Get the software image name.

```
qsc-cli chip-software list-workspace
```

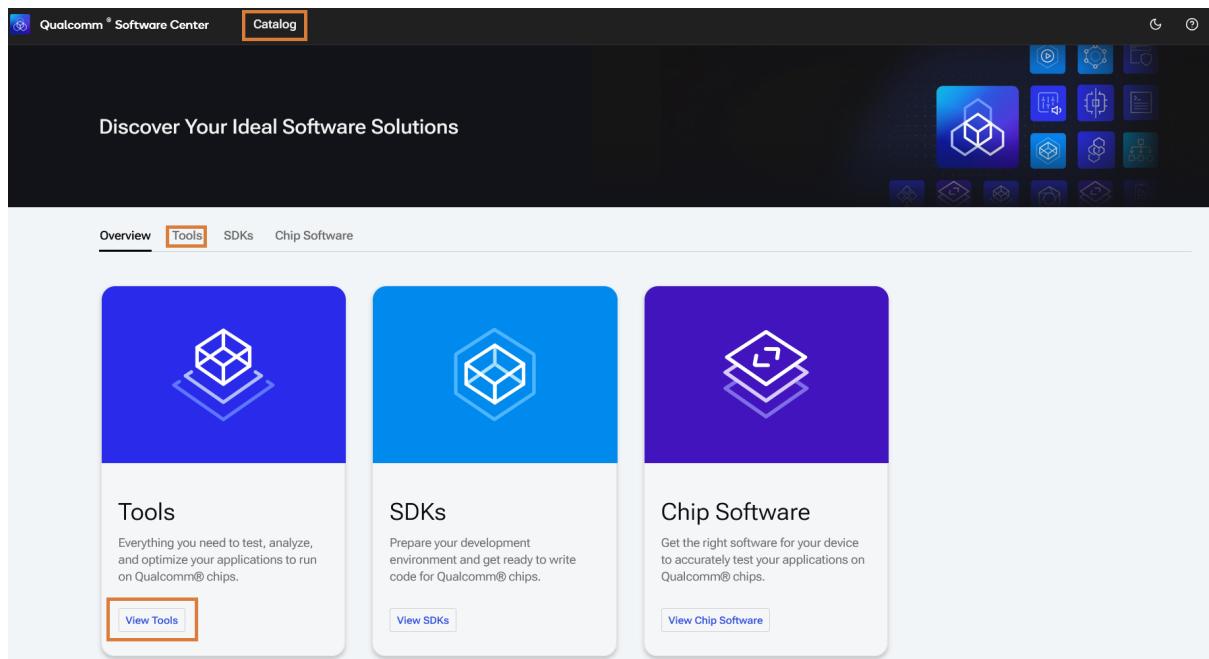
In output, look for *Compilable Images* for the software image names.

# 6 Tools and SDKs

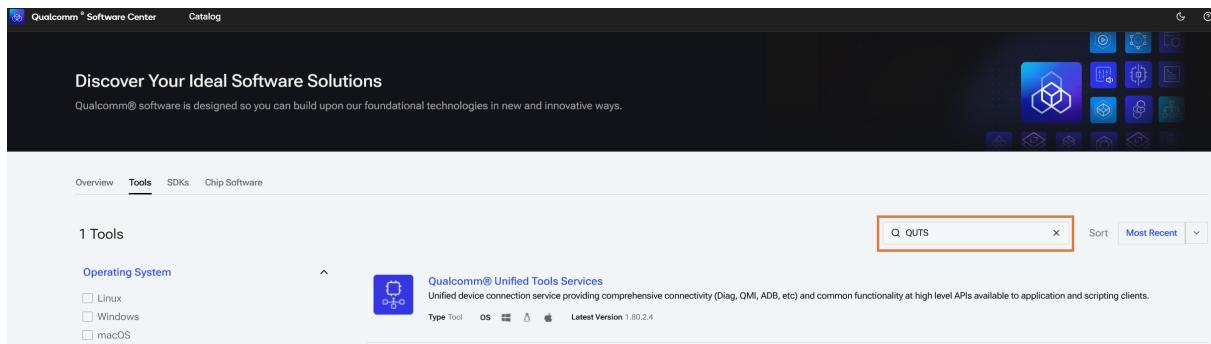
---

## 6.1 Find a tool or SDK

1. In the Qualcomm Software Center web [portal](#) (or desktop application), navigate to **Catalog** on the top menu bar and select the **Tools** category.
2. The Tools page lists tools to which you have access. Click on the card for a tool to navigate to the tool detail page.



3. When you know name of the tool you are looking for, you can search in the menu bar.



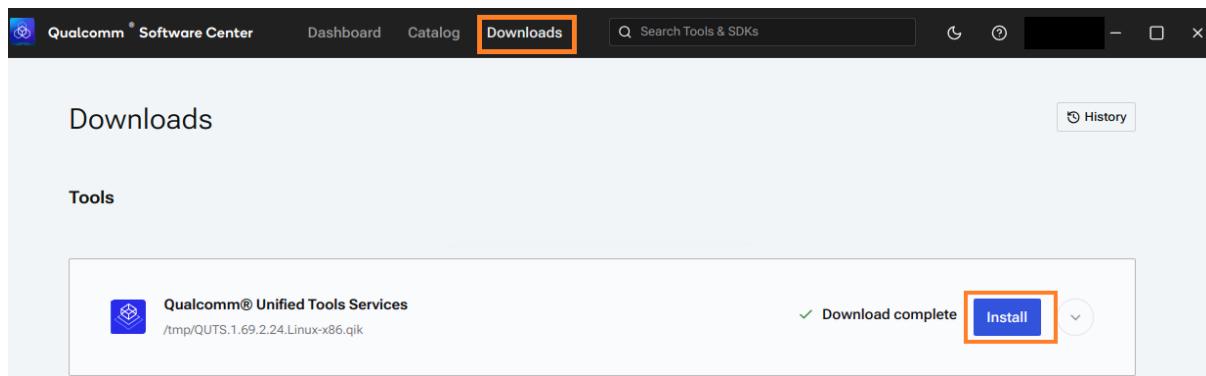
## 6.2 Install a tool or SDK

1. Select the relevant tool version (by default, the latest version is selected) on the tool detail page and click **Install**.

The screenshot shows the tool detail page for the Qualcomm Unified Tools Service. On the left, there's a sidebar with 'Installations' and 'Licenses' sections. The main area shows 'OS Details' (Type: Linux, Architecture: x86, Distribution: Debian), a note about Debian coverage, 'Tool Details' (Version: 1.73.2.26, Release Date: 6/27/2024), and a large 'Install' button at the bottom right.

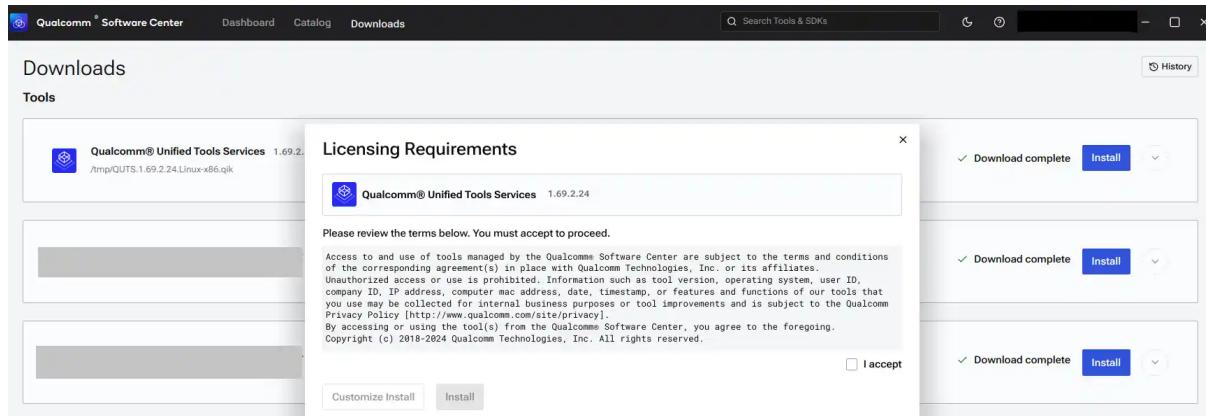
This step automatically activates entitlements (the Site type license groups) for the tool.

2. When the download is complete, navigate to the downloads page (from the notification pop up or top menu bar) and click **Install** to start installation.

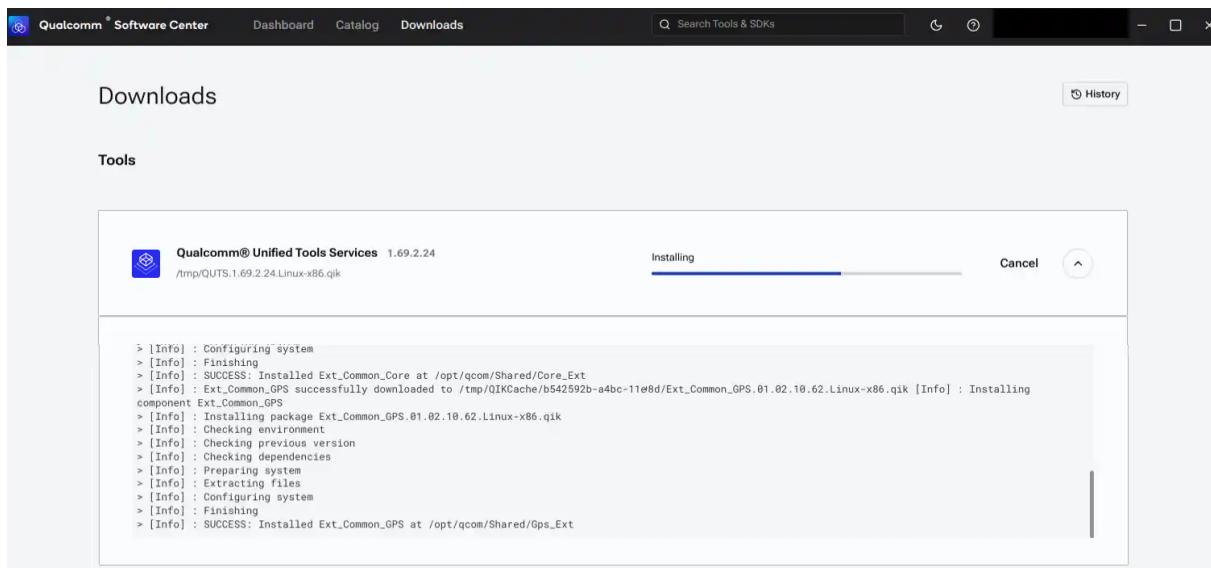


The license agreement displays.

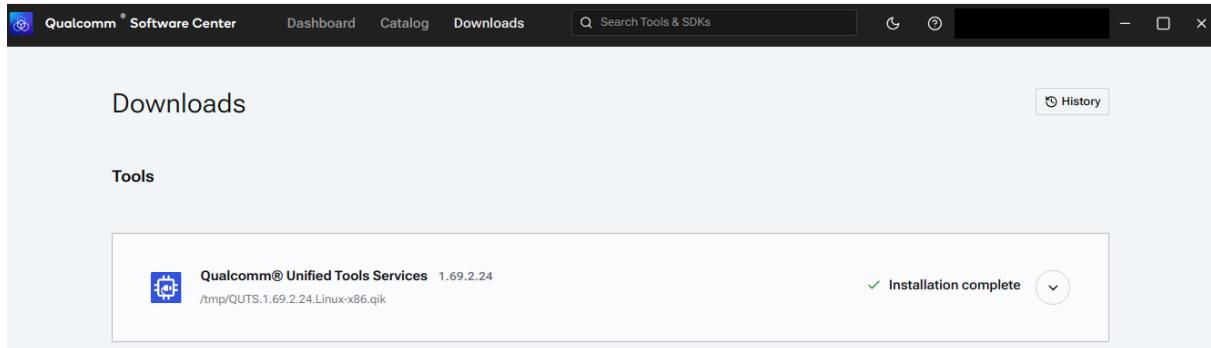
- Review the license agreement and select **I accept** to proceed with the installation.



- To view the installation progress and summary, expand the panel for the tool in the **Downloads** page.



View the installed tool version via the tool detail page or the **Installations** section on the dashboard page.

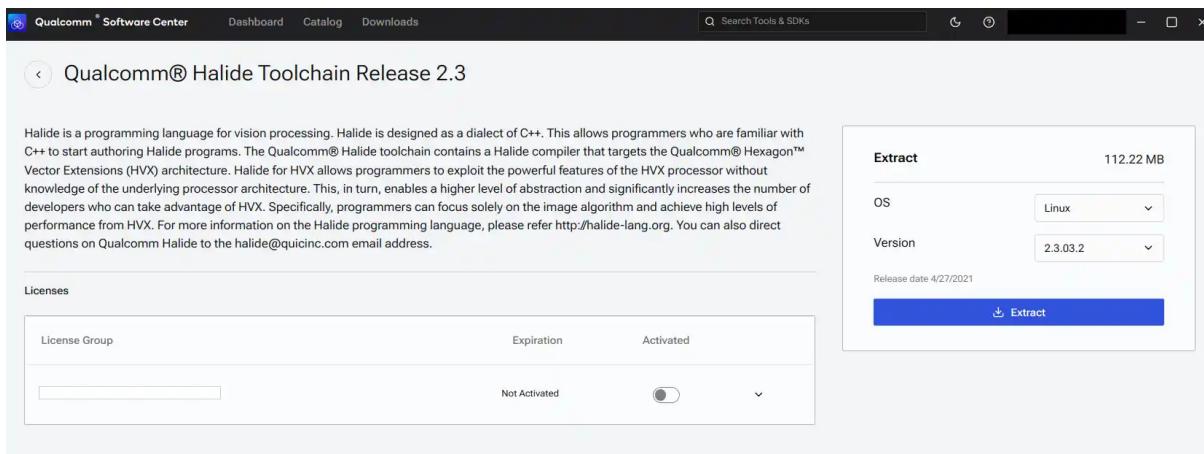


5. To launch the tool after it installs, search for the tool in the applications menu on the machine.

---

**Note:** The QSC installs and extracts tools of the type **Extract**, but does not track the Extract type tools as “installed”.

---



## 6.3 Manage licenses

Most Qualcomm tools require download and install of licenses from the Qualcomm Software Center (QSC). Site, Purchase, Demo, and Factory license types are provided based on user entitlement to the tool and features associated with the tool.

- Site licenses are provided as part of your company agreement with Qualcomm for specific chipsets. These Site licenses are visible when you click on a particular tool and automatically activated when you install them, requiring no user action for license activation.
- Purchase licenses are provided as part of your purchase order for a tool via the Qualcomm Sales Center (<https://cp.qti.qualcomm.com>). Purchase licenses must be manually activated in the Qualcomm Software Center desktop application to install the purchased tool.
- Demo licenses are provided for trial purposes or via an agreement with Qualcomm. These licenses are received via your account sales or technical account manager. Similar to purchase licenses, demo licenses must be manually activated in the Qualcomm Software Center desktop application.
- Factory licenses are entitled to customers involved in manufacturing and are governed by agreements with Qualcomm and are intended for factory floor machines without an Internet connection.

## 6.4 Use the QSC CLI

**Note:** qpm-cli commands are now qsc-cli commands.

These commands are applicable for version 1.13.x onwards.

The Qualcomm Software Center qsc-cli command line interface can be used to activate a license, and download and install tools and SDKs. qsc-cli is installed with QSC.

## Help

Display information such as installed version and available versions for the tool or SDK specified as input.

```
qsc-cli tool --help
```

or

```
qsc-cli sdk --help
```

## Log in

Log in using your Qualcomm ID.

```
qsc-cli login -u <your username>
```

## Manage tools and SDKs

Display a list of tools or SDKs to which the logged in user has access.

```
qsc-cli tool info
```

or

```
qsc-cli sdk info
```

Display tool or SDK information such as installed version and available versions for the tool specified as input.

```
qsc-cli tool info --name <tool name>
```

Activate entitlements, i.e., SITE license groups, using a tool or SDK name as input. You can also activate licenses using a specific licensegroup UUID as input.

```
qsc-cli tool activate-license --name <tool name>
```

or

```
qsc-cli tool activate-license --license-id <license group ID>
```

Specify a tool or SDK to download and install.

```
qsc-cli tool install --name <tool name>
```

For more options, refer to `qsc-cli tool install --help`.

Extract type tools.

```
qsc-cli tool extract --name <tool name>
```

For more options, refer to `qsc-cli tool extract --help`.

## Install PCAT and QUD

Run the following command to install PCAT.

```
qsc-cli tool install --name pcat --activate-default-license
```

Run the following command to install QUD.

```
qsc-cli tool install --name qud --activate-default-license
```

## 6.5 Deprecated qpm-cli commands

The following commands are for reference only.

### Help

Display the qpm-cli help menu for options and updates.

```
qpm-cli -h
```

### Log in

Log in using your Qualcomm ID.

```
qpm-cli --login -u <your username>
```

## Manage tools and SDKs

Display a list of tools or SDKs to which the logged in user has access.

```
qpm-cli --product-list
```

Display tool or SDK information such as installed version and available versions for the tool specified as input.

```
qpm-cli --info [ProductName]
```

Use the tool or SDK name to activate the license and activate entitlements (SITE license groups for the tool or SDK) or licensegroup UUID.

```
qpm-cli --license-activate [ProductName or license group UUID]
```

Specify a tool or SDK to download and install.

```
qpm-cli --install [ProductName or full path to downloaded <product>.qik installer file]
```

Extract type tools.

```
qpm-cli --extract [ProductName or the full path to downloaded <product>.qik installer file]
```

## Install PCAT and QUD

Run the following commands to use the QPM CLI to install PCAT and QUD.

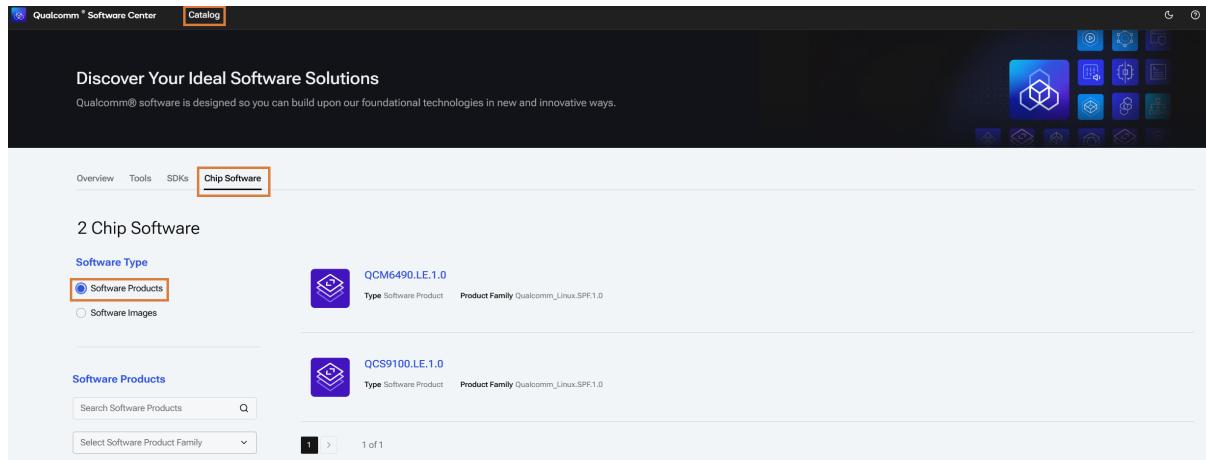
```
qpm-cli --login  
qpm-cli --install pcat --activate-default-license  
qpm-cli --install qud --activate-default-license
```

# 7 Chip software

## 7.1 Download software product

On the Qualcomm Software Center web [portal](#) (or desktop application), navigate to **Catalog** on the top menu bar and select the **Chip Software** category.

Select **Software Products** to see all software products (SPs) and software product families (SPFs) to which you have access. Click on a row to navigate to the software product detail page.



The screenshot shows the Qualcomm Software Center interface. At the top, there's a navigation bar with 'Qualcomm® Software Center' and 'Catalog' selected. Below the bar, a banner says 'Discover Your Ideal Software Solutions' with a subtext about Qualcomm software being designed for innovation. The main content area has a heading '2 Chip Software'. Under 'Software Type', 'Software Products' is selected (indicated by a red box). There are two entries: 'QCM6490.LE.1.0' and 'QCS9100.LE.1.0', both listed as 'Type Software Product' under 'Product Family Qualcomm\_Linux.SPF.1.0'. Each entry includes a small icon and a link. At the bottom left, there's a search bar and a dropdown menu for selecting a software product family. The bottom right shows a navigation bar with icons for back, forward, and search.

Select a distribution to see the releases and corresponding Git commands for the selected release.

Distributions		Search Distribution	
Distribution		Release Date	
<a href="#">Qualcomm_Linux.SPF.1.0 AMSS Standard OEM </a> >		04-25-2025	<a href="#">29 Releases</a> ^
This distro is for the Standard OEM customers with source distribution.			
Release Name	Release Tag	Release Date	Compile Status
QCM6490.LE.1.0-00355-STD.PROD-1	r1.0_00083.0	04-25-2025	Compilation Required >
QCM6490.LE.1.0-00354-STD.PROD-2	r1.0_00078.0	04-05-2025	Compilation Required >
QCM6490.LE.1.0-00349-STD.PROD-1	r1.0_00075.0	03-31-2025	Compilation Required >

Git CLI commands can be used to download SP releases. Download proprietary software via the qpm-git.qualcomm.com repo(s). Download open source software via CodeLinaro.

QCM6490.LE.1.0-00149-STD.PROD-1

[r1.0\\_00018.0](#) Choose a specific Release Tag, or select a Branch to view the latest Release on that branch

Release Tag r1.0_00018.0	Release Date 12-03-2024 11:15:54	Distro Name Qualcomm_Linux.SPF.1.0 AMSS Standard OEM
ECCN 5D002.c.1	Serial Number 378425	

**Launcher**  
A new way to download and compile all relevant resources for your device to get up and running fast.  
[Start Launcher](#)

**Git CLI**

```
git clone https://qpm-git.qualcomm.com/home2/git/qualcomm/qualcomm-linux-spf-1-0_amss_standard_oem.gi: □
git clone https://qpm-git.qualcomm.com/home2/git/qualcomm/qualcomm-linux-spf-1-0_amss_standard_oem.gi: □
```

Clone release

```
git clone -b r1.0_00018.0 --depth 1 https://qpm-git.qualcomm.com/home2/git/qualcomm/qualcomm-linux-sp □
```

**CodeLinaro**

For the SPs that Launcher supports, it is highly recommended to use the [Launcher](#) workflow on the Qualcomm Software Center desktop application to simplify the download and compile process.

The screenshot shows the Qualcomm Software Center web interface. At the top, there's a navigation bar with links for Dashboard, Catalog, and Downloads, along with a search bar and user icons. Below the navigation bar, the page title is "Qualcomm\_Linux.SPF.1.0". A descriptive text states: "SP will follow agile development methodology. This SP is created to support multiple customers from Camera, Drones, Robotics, ...". Underneath, there's a section titled "Download Options" with three choices: "Launcher" (represented by a blue icon), "yocto PROJECT" (represented by a blue icon), and "GIT CLI" (represented by a blue icon). Each option has a brief description and a link: "Start Launcher >" for the launcher, "Download with Yocto >" for the yocto project, and "Show More >" for the GIT CLI.

## 7.2 Download software image

On the Qualcomm Software Center web [portal](#) (or desktop application), navigate to **Catalog** on the top menu bar and select the **Chip Software** category.

Select **Software Images** to see all independent software images (SIs) to which you have access.

The screenshot shows the Qualcomm Software Center web interface with the "Catalog" tab selected. Below it, the "Chip Software" category is highlighted with an orange border. The main content area displays a grid of software solutions. One solution is selected, showing its details: "LA.QSSI.15.0" (Type: Software Image). Below this, another solution is listed: "LA.QSSI.16.0" (Type: Software Image). At the bottom of the list, there's a search bar labeled "Search Software Images" and a pagination indicator showing "1 of 1".

Select a distribution to see the releases and corresponding Git commands for the selected release.

The screenshot shows the Qualcomm Software Center interface. At the top, there's a navigation bar with the Qualcomm logo, "Qualcomm Software Center", and a "Catalog" link. Below the navigation bar, the title "LA.QSSI.16.0" is displayed. Under the title, there's a section titled "Distributions". A table lists several distributions:

Distribution	Release Date	Releases
<a href="#">LA.QSSI.16.0 AP OEM OSH &gt;</a> Qualcomm Single System Image Open Source Software APSS code for Android W	28-03-2025	<a href="#">3 Releases ^</a>
Image Build	Release Tag	Release Date
LA.QSSI.16.0-14000-qssi.0-1	r14000.1	28-03-2025 >
LA.QSSI.16.0-12800-qssi.0-1	r12800.1	28-03-2025 >
LA.QSSI.16.0-11700-qssi.0-1	r11700.1	28-03-2025 >
<a href="#">LA.QSSI.16.0 AP OEM PROP &gt;</a> APSS proprietary source code with commit history - APSS images only	28-03-2025	<a href="#">3 Releases ^</a>
<a href="#">LA.QSSI.16.0 HLOS DEV   &gt;</a> Developer distro for HLOS-related images in SRC to support development	28-03-2025	<a href="#">3 Releases ^</a>

Git CLI commands can be used to download software image releases. Download proprietary software via the qpm-git.qualcomm.com repo(s). Download open source software via CodeLinaro.

The screenshot shows the details for the LA.QSSI.16.0|AP|OEM|OSH distribution. It includes fields for "Release" (r14000.1), "Release Tag" (r14000.1), "Release Date" (28-03-2025 02:15:29), "Distribution Name" (LA.QSSI.16.0|AP|OEM|OSH), "ECCN" (5D002.c.1), and "Serial Number" (7).

Below this, there are two sections: "GIT CLI" and "CodeLinaro".

**GIT CLI**

```
git clone https://qpm-git.qualcomm.com/home2/git/qualcomm/la-qssi-16-0_ap_oem_osh.git
```

Clone release

```
git clone -b r14000.1 --depth 1 https://qpm-git.qualcomm.com/home2/git/qualcomm/la-qssi-16-0_ap_oem_osh.git
```

**CodeLinaro**

Clone repo

```
repo init --depth=1 --current-branch -u https://git.codelinaro.org/clo/private/la-qssi-16-0/la/system/manifest.git -b release
```

# 8 Troubleshooting

---

## 8.1 Log in issues

To access the Qualcomm Software Center (web [portal](#) and desktop application), you need a Qualcomm ID. Sign up at <https://myaccount.qualcomm.com/signup>.

If you have log in issues using correct Qualcomm ID credentials with the Qualcomm Software Center desktop application, see [firewall requirements](#) to ensure that connectivity functions as expected between your network and Qualcomm servers.

## 8.2 Firewall requirements

The QSC server requires access to the following URLs:

- <https://apigwx-aws.qualcomm.com> (port 443)
- <https://d1fmewm8udj8en.cloudfront.net> (port 443)
- <https://dvbe2v39djgwg.cloudfront.net/> (port 443)
- <https://d1k9x1hli7v4r5.cloudfront.net/> (port 443)
- (Optional, recommended for users based in China)  
<https://qpm.chinacloudfront.qualcomm.com> (port 443)

Run the following commands to check DNS and firewall issues.

1. Verify that the client can resolve the API endpoint via DNS.

```
nslookup apigwx-aws.qualcomm.com
```

**Sample output**

```
user@host:~$ nslookup apigwx-aws.qualcomm.com
Server:      XXX.XX.XXX.X
Address:     XXX.XX.XXX.X#XX

apigwx-aws.qualcomm.com canonical name = apigwx-aws.gtpgeo.com.
apigwx-aws.gtpgeo.com   canonical name = d2uf15avqx5eaz.cloudfront.net.
Name:        d2uf15avqx5eaz.cloudfront.net
Address:    XX.XXX.XXX.XXX
Name:        d2uf15avqx5eaz.cloudfront.net
```

2. Check the connection.

```
curl -v https://apigwx-aws.qualcomm.com
```

**Sample output**

```
Rebuilt URL to: https://apigwx-aws.qualcomm.com/
timeout on name lookup is not supported
Trying x.x.x.x...
TCP_NODELAY set
Connected to apigwx-aws.qualcomm.com (x.x.x.x) port
443 (#0)
schannel: SSL/TLS connection with
apigwx-aws.qualcomm.com port 443 (step 1/3)
schannel: checking server certificate revocation
...
schannel: SSL/TLS handshake complete
schannel: SSL/TLS connection with
apigwx-aws.qualcomm.com port 443 (step 3/3)
schannel: stored credential handle in session cache
GET / HTTP/1.1
Connection #0 to host apigwx-aws.qualcomm.com left intact
```

## Troubleshoot Linux clients

Test these steps in the following order. If a step is completed successfully, proceed to the next test.

1. Verify that the client can resolve the API endpoint via DNS.

- In a Linux terminal, type nslookup and click **Enter**.
- Type the API endpoint apigwx-aws.qualcomm.com and click **Enter**.

The expected response is a list of IPs similar to the following image.

```
user@host:~$ nslookup
> apigwx-aws.qualcomm.com
Server:      172.26.240.1
Address:     172.26.240.1#53

apigwx-aws.qualcomm.com canonical name = apigwx-aws.gtpgeo.com.
apigwx-aws.gtpgeo.com   canonical name = d2uf15avqx5eaz.cloudfront.net.
Name:        d2uf15avqx5eaz.cloudfront.net
Address:    XX.XXX.XXX.XXX
>
```

If no IPs are returned, contact your company DNS team to troubleshoot resolving apigwx-aws.qualcomm.com from your client or issues with your client DNS.

2. Verify that the client can reach the API endpoint via Ping.

- In a Linux terminal type ping apigwx-aws.qualcomm.com.
- Type Ctrl + C to stop the ping test after five to ten seconds.

The following image shows the response with zero packet loss.

```
user@host:~$ ping apigwx-aws.qualcomm.com
PING d2uf15avqx5eaz.cloudfront.net (XX.XXX.XXX.XXX) 56(84) bytes of data.
64 bytes from server-XX.XXX.XXX.XXX.lax50.r.cloudfront.net (XX.XXX.XXX.XXX): icmp_seq=1 ttl=248 time=14.6 ms
64 bytes from server-XX.XXX.XXX.XXX.lax50.r.cloudfront.net (XX.XXX.XXX.XXX): icmp_seq=2 ttl=248 time=15.4 ms
64 bytes from server-XX.XXX.XXX.XXX.lax50.r.cloudfront.net (XX.XXX.XXX.XXX): icmp_seq=3 ttl=248 time=17.5 ms
^C
--- d2uf15avqx5eaz.cloudfront.net ping statistics ---
4 packets transmitted, 3 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 14.582/15.840/17.529/1.240 ms
```

If packets are lost or not received, contact your networking team to troubleshoot apigwx-aws.qualcomm.com via Ping.

3. Verify that the client can reach the API endpoint via Telnet.

- In a Linux terminal, type `telnet apigwx-aws.qualcomm.com 443`.  
The expected display is a Connected to \*\*\*\*\*.cloudfront.net message.
- Type Ctrl + C to close the connection.

For connection failed or timed out messages, contact your network team to troubleshoot `apigwx-aws.qualcomm.com` via Telnet from your client.

## Troubleshoot Windows clients

Test these steps in order. If a step is completed successfully, move to the next test.

### 1. Verify that the client can resolve the API endpoint via DNS

- Open a windows command prompt.
- Type `nslookup` and click **Enter**.
- Type the API endpoint [apigwx-aws.qualcomm.com](http://apigwx-aws.qualcomm.com) and click Enter.

The expected response is a list of IPs similar to the following image.

```
C:\Windows\System32> nslookup
Default Server: san-qrc-dcr-dns-02.qualcomm.com
Address: XX.XX.XX.XX

> apigwx-aws.qualcomm.com
Server: san-qrc-dcr-dns-02.qualcomm.com
Address: XX.XX.XX.XX

Name: d2uf15avqx5eaz.cloudfront.net
Addresses: XX.XXX.XXX.XXX
           XX.XXX.XXX.XXX
           XX.XXX.XXX.XXX
           XX.XXX.XXX.XXX
Aliases: apigwx-aws.qualcomm.com
          apigwx-aws.gtpgeo.com

>
```

If no IPs are returned, contact your company DNS team to troubleshoot `apigwx-aws.qualcomm.com` not resolvable from your client or any issue with your client DNS.

### 2. Verify that the client can reach the API endpoint via Ping

- Open a Windows command prompt.
- Type `ping apigwx-aws.qualcomm.com`.

The expected response is a display with zero packets lost as shown in the following image.

```
C:\Windows\System32>ping apigwx-aws.qualcomm.com

Pinging d2uf15avqx5eaz.cloudfront.net [XX.XXX.XXX.X] with 32 bytes of data:
Reply from XX.XXX.XXX.X: bytes=32 time=16ms TTL=249
Reply from XX.XXX.XXX.X: bytes=32 time=31ms TTL=249
Reply from XX.XXX.XXX.X: bytes=32 time=16ms TTL=249
Reply from XX.XXX.XXX.X: bytes=32 time=15ms TTL=249

Ping statistics for 18.154.144.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 15ms, Maximum = 31ms, Average = 19ms

C:\Windows\System32>
```

For packets lost or not received, contact your networking team to troubleshoot `apigwx-aws.qualcomm.com` via Ping.

### 3. Verify that the client can reach the API endpoint via Tracert

- Open a windows command prompt.
- Type `tracert apigwx-aws.qualcomm.com`.

The expected response is an initial Tracing route to `XXXXXX.cloudfront.net` [IP address] message.

The trace route shows the internet route taken to reach the endpoint from [Step 2](#).

The following is an example of the expected response.

```
C:\Windows\System32>tracert apigwx-aws.qualcomm.com

Tracing route to d2uf15avqx5eaz.cloudfront.net [XX.XXX.XXX.X]
over a maximum of 30 hops:

 1  291 ms  284 ms  283 ms  san-qrc004-ras-vpn-05-inside.qualcomm.com [XX.XXX.XXX.X]
 2  279 ms  293 ms  286 ms  SAN-RAS-FW-01-RAS-GW.qualcomm.com [XX.XXX.XXX.X]
 3  284 ms  283 ms  288 ms  san-r108-tra-rt-01-vlan208.qualcomm.com [XX.XXX.XXX.X]
 4  289 ms  284 ms  465 ms  san-r108-tra-rt-02_to_san-r108-tra-rt-01.qualcomm.com [XX.XXX.XXX.X]
 5  297 ms  554 ms  291 ms  qf-pa-inet-inside.qualcomm.com [XX.XXX.XXX.X]
 6  293 ms  288 ms  286 ms  san-r-dmz-rt-03-v192.qualcomm.com [XX.XXX.XXX.X]
 7  286 ms  289 ms  289 ms  san-r-bdr-rt-01_to_san-r-dmz-rt-03.qualcomm.com [XX.XXX.XXX.X]
 8  291 ms  295 ms  291 ms  206.72.211.146.any2ixcoresite.com [XX.XXX.XXX.X]
 9  *       *       *       Request timed out.
10  *       *       *       Request timed out.
11  289 ms  293 ms  301 ms  150.222.232.118
12  *       *       *       Request timed out.
13  *       *       *       Request timed out.
14  *       *       *       Request timed out.
15  *       *       *       Request timed out.
16  *       *       *       Request timed out.
17  *       *       *       Request timed out.
18  *       *       *       Request timed out.
19  *       *       *       Request timed out.
20  *       *       *       Request timed out.
21  291 ms  288 ms  *       150.222.232.118
22  1037 ms  919 ms  867 ms  server-18-154-144-8.lax50.r.cloudfront.net [XX.XXX.XXX.X]

Trace complete.
```

Some timeouts can be expected.

If there are connection issues such as the final hop does not reach the endpoint from **Step 2**, there is an issue with the client route to the `apigwx-aws.qualcomm.com` endpoint. Contact your IT group with an example of the *client cannot route to the apigwx-aws.qualcomm.com Qualcomm endpoint* message.

## 8.3 Contact support

For technical issues, visit the [Support site](#).

For assistance or clarification on information in this document, open a technical support case at <https://support.qualcomm.com/>. You will need to register for a Qualcomm ID account and your company must have support enabled to access the case system.

To file a support case to report an issue, include the following information:

- Host operating system details
- Version of the Qualcomm Software Center desktop application
- Details of the issue with exact steps to reproduce
- Relevant UI screenshot(s)
- Logs

To collect logs on the Qualcomm Software Center desktop application, go to the Help menu > **Download Logs** > Select the date range > **Download**.

---

**Note:** Logs can be collected manually from the locations listed below.

---

### Linux

- `/var/tmp/qcom/softwarecenter/logs`
- `/var/tmp/qcom/qsccli/logs`
- `/var/tmp/qcom/qpmcli/logs`

### Windows

- `C:\ProgramData\Qualcomm\SoftwareCenter\Logs`
- `%temp%\QIK\Logs`
- `C:\Windows\Temp\QIK\Logs`
- `%temp%\Qualcomm\QPMCLI\Logs`

---

*Troubleshooting*

Other systems and support resources are listed on <https://qualcomm.com/support>. For further assistance, send an email to [qualcomm.support@qti.qualcomm.com](mailto:qualcomm.support@qti.qualcomm.com).

## LEGAL INFORMATION

Your access to and use of this material, along with any documents, software, specifications, reference board files, drawings, diagnostics and other information contained herein (collectively this "Material"), is subject to your (including the corporation or other legal entity you represent, collectively "You" or "Your") acceptance of the terms and conditions ("Terms of Use") set forth below. If You do not agree to these Terms of Use, you may not use this Material and shall immediately destroy any copy thereof.

### 1) Legal Notice.

This Material is being made available to You solely for Your internal use with those products and service offerings of Qualcomm Technologies, Inc. ("Qualcomm Technologies"), its affiliates and/or licensors described in this Material, and shall not be used for any other purposes. If this Material is marked as "**Qualcomm Internal Use Only**", no license is granted to You herein, and You must immediately (a) destroy or return this Material to Qualcomm Technologies, and (b) report Your receipt of this Material to [qualcomm.support@qti.qualcomm.com](mailto:qualcomm.support@qti.qualcomm.com). This Material may not be altered, edited, or modified in any way without Qualcomm Technologies' prior written approval, nor may it be used for any machine learning or artificial intelligence development purpose which results, whether directly or indirectly, in the creation or development of an automated device, program, tool, algorithm, process, methodology, product and/or other output. Unauthorized use or disclosure of this Material or the information contained herein is strictly prohibited, and You agree to indemnify Qualcomm Technologies, its affiliates and licensors for any damages or losses suffered by Qualcomm Technologies, its affiliates and/or licensors for any such unauthorized uses or disclosures of this Material, in whole or part.

Qualcomm Technologies, its affiliates and/or licensors retain all rights and ownership in and to this Material. No license to any trademark, patent, copyright, mask work protection right or any other intellectual property right is either granted or implied by this Material or any information disclosed herein, including, but not limited to, any license to make, use, import or sell any product, service or technology offering embodying any of the information in this Material.

THIS MATERIAL IS BEING PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, WHETHER EXPRESSED, IMPLIED, STATUTORY OR OTHERWISE. TO THE MAXIMUM EXTENT PERMITTED BY LAW, QUALCOMM TECHNOLOGIES, ITS AFFILIATES AND/OR LICENSORS SPECIFICALLY DISCLAIM ALL WARRANTIES OF TITLE, MERCHANTABILITY, NON-INFRINGEMENT, FITNESS FOR A PARTICULAR PURPOSE, SATISFACTORY QUALITY, COMPLETENESS OR ACCURACY, AND ALL WARRANTIES ARISING OUT OF TRADE USAGE OR OUT OF A COURSE OF DEALING OR COURSE OF PERFORMANCE. MOREOVER, NEITHER QUALCOMM TECHNOLOGIES, NOR ANY OF ITS AFFILIATES AND/OR LICENSORS, SHALL BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY EXPENSES, LOSSES, USE, OR ACTIONS HOWSOEVER INCURRED OR UNDERTAKEN BY YOU IN RELIANCE ON THIS MATERIAL.

Certain product kits, tools and other items referenced in this Material may require You to accept additional terms and conditions before accessing or using those items.

Technical data specified in this Material may be subject to U.S. and other applicable export control laws. Transmission contrary to U.S. and any other applicable law is strictly prohibited.

Nothing in this Material is an offer to sell any of the components or devices referenced herein.

This Material is subject to change without further notification.

In the event of a conflict between these Terms of Use and the *Website Terms of Use* on [www.qualcomm.com](http://www.qualcomm.com), the *Qualcomm Privacy Policy* referenced on [www.qualcomm.com](http://www.qualcomm.com), or other legal statements or notices found on prior pages of the Material, these Terms of Use will control. In the event of a conflict between these Terms of Use and any other agreement (written or click-through, including, without limitation any non-disclosure agreement) executed by You and Qualcomm Technologies or a Qualcomm Technologies affiliate and/or licensor with respect to Your access to and use of this Material, the other agreement will control.

These Terms of Use shall be governed by and construed and enforced in accordance with the laws of the State of California, excluding the U.N. Convention on International Sale of Goods, without regard to conflict of laws principles. Any dispute, claim or controversy arising out of or relating to these Terms of Use, or the breach or validity hereof, shall be adjudicated only by a court of competent jurisdiction in the county of San Diego, State of California, and You hereby consent to the personal jurisdiction of such courts for that purpose.

### 2) Trademark and Product Attribution Statements.

Qualcomm is a trademark or registered trademark of Qualcomm Incorporated. Arm is a registered trademark of Arm Limited (or its subsidiaries) in the U.S. and/or elsewhere. The Bluetooth® word mark is a registered trademark owned by Bluetooth SIG, Inc. Other product and brand names referenced in this Material may be trademarks or registered trademarks of their respective owners.

Snapdragon and Qualcomm branded products referenced in this Material are products of Qualcomm Technologies, Inc. and/or its subsidiaries. Qualcomm patented technologies are licensed by Qualcomm Incorporated.