Embedded Platforms

You can develop applications for the following embedded platforms:

- Bare Metal
- Boot2Qt
- Generic Remote Linux
- Microcontroller Units (MCU)
- QNX

You must install the tool chain for building applications for the targeted embedded platform on the development PC and use the Qt Maintenance Tool to install Qt libraries that are built for the platform. You can then add a kit with the tool chain and the Qt version for the device's architecture. When possible, the Maintenance Tool creates suitable kits for you.

You can connect embedded devices to the development PC to run, debug, and analyze applications built for them from Qt Creator.

Bare Metal

You can run and debug applications on small devices that are not supported by the generic remote Linux device plugin by using GDB or a hardware debugger.

For more information about developing applications for Bare Metal devices, see Connecting Bare Metal Devices.

Boot2Qt

The Boot2Qt stack can be made to run on a variety of hardware. For license holders, tooling is provided to customize the contents of the stack as well as to take it into desired production hardware.

Either Windows 10 64-bit or later or Ubuntu Linux 64-bit 20.04 LTS or later is required to install and use Boot2Qt.

The following topics contain more information about developing applications for Boot2Qt devices:

- Boot2Qt: Supported Target Devices and Development Hosts
- Boot2Qt: Installation Guides
- Connecting Boot2Qt Devices
- Specifying Run Settings for Boot2Qt Devices
- Deploying Applications to Boot2Qt Devices

Generic Remote Linux

You must have a tool chain for building applications for embedded Linux devices installed on the development PC.

The following topics contain more information about developing applications for generic remote Linux devices:

- Connecting Generic Remote Linux Devices
- Deploying Applications to Generic Remote Linux Devices
- Specifying Run Settings for Linux-Based Devices
- Running on Generic Remote Linux Devices
- Qt Creator Plugin for Qt Application Manager

Microcontroller Units (MCU)

You need the GNU Arm Embedded GCC compiler, libraries, and other GNU tools necessary for bare metal software development on devices based on the Arm Cortex-M processors.

The following topics contain more information about developing applications for MCUs:

- Connecting MCUs
- Running Applications on MCUs
- Qt for MCUs

QNX

The QNX Neutrino RTOS should provide a few additional command line tools and services, as described in Qt for QNX.

> **Note:** In Qt 6, Qt Creator support for QNX is considered experimental.

The following topics contain more information about developing applications for QNX devices:

- Connecting QNX Devices
- Deploying Applications to QNX Neutrino Devices
- Specifying Run Settings for QNX Devices
- Running on QNX Devices
- Qt for QNX

‹ Desktop Platforms

Mobile Platforms ›

Qt DOCUMENTATION

The Qt Company

Contact Us

Company

About Us
Investors
Newsroom
Careers
Office Locations

Licensing

Terms & Conditions
Open Source
FAQ

Support

Support Services
Professional Services
Partners
Training

For Customers

Support Center
Downloads
Qt Login
Contact Us
Customer Success

Community

Contribute to Qt
Forum
Wiki
Downloads
Marketplace

Feedback Sign In