

[Qt 创建者手册](#) > [连接 iOS 设备](#)

连接 iOS 设备

您可以使用USB电缆将iOS设备连接到本地机器，以运行Qt Creator为其构建的应用程序。

为了能够在macOS上使用Qt Creator，您必须安装Xcode，因此，您已经拥有为iOS构建应用程序的工具链。Qt Creator会自动检测工具链并创建必要的**工具包**，以便为配置的iOS设备构建应用程序并在上运行它们。

您只需要为iOS构建的Qt库。您可以将它们作为 Qt 5.2 或更高版本的一部分进行安装。

配置设备

Qt Creator 与 iOS 设备之间的连接通过使用您在[注册 Apple 开发者计划](#)时从 Apple 收到的证书来保护。配置设备时，证书将复制到设备。

首次将设备连接到本地计算机时，系统会要求您在设备上启用开发人员模式。下次连接设备时，Qt Creator 会自动检测到它。若要禁用与不用于开发的设备的自动连接，请选择“**首选项**”> **iOS**，然后取消选中“**询问未处于开发人员模式的设备**”复选框。

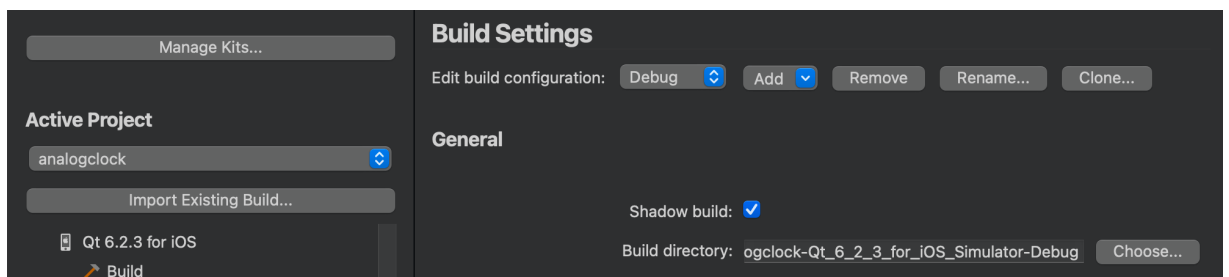
注意：配置设备和 UI 的过程会根据您使用的 Xcode 版本而略有不同。我们建议您使用最新的可用 Xcode 版本。

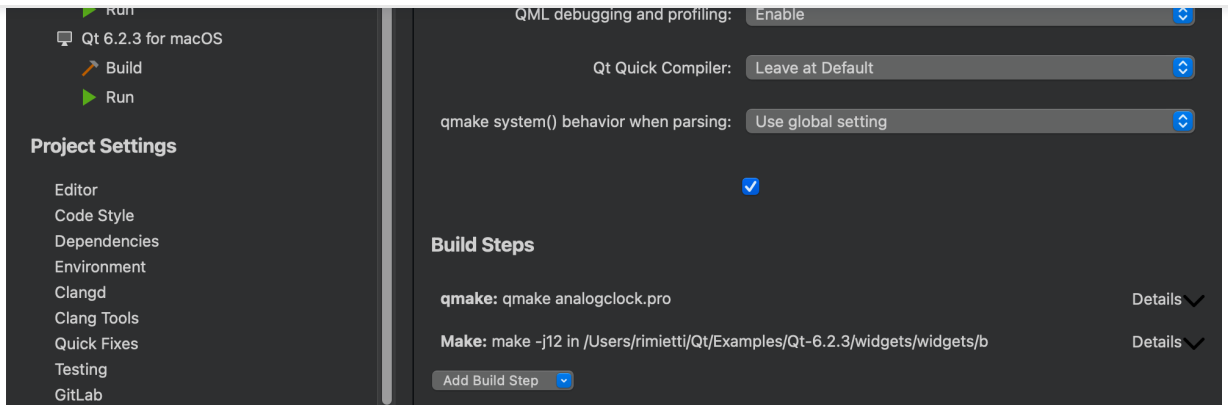
要配置 Qt 创建者与 iOS 设备之间的连接：请执行以下操作：

1. 确保您已安装适用于 iOS 的 Xcode 和 Qt。
2. 使用 USB 电缆将设备连接到本地计算机。
3. 启动 Xcode 以配置设备。

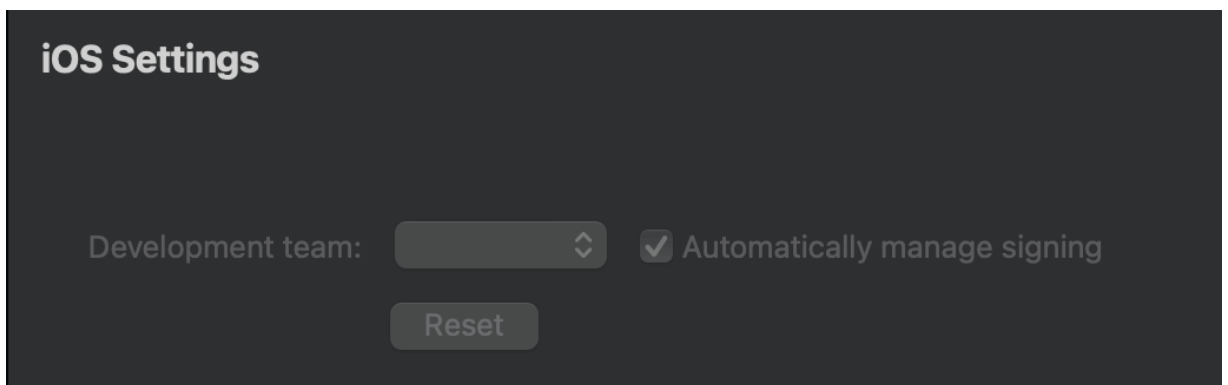
例如，在 Xcode 版本 7.3.0 中，选择“**窗口**>**设备**> + >**添加设备**”以添加连接的设备。

4. 指定生成设置：
 1. 为要为设备开发的应用程序打开一个项目。
 2. 选择“**项目**>**生成 & 运行**”以选择用于在 iOS 上生成和运行应用程序的应用程序的**工具包**。





3. 在“iOS 设置”中，选择要用于对应用程序进行签名和预配的开发团队。您必须使用 Apple 开发人员帐户在 Xcode 中配置开发团队和配置配置文件。



4. Select the **Automatically manage signing** check box to automatically select the provisioning profile and signing certificate on your local machine that matches the entitlements and the bundle identifier of the iOS device.

5. Select **Run** to specify run settings.

Usually, you can use the default settings.

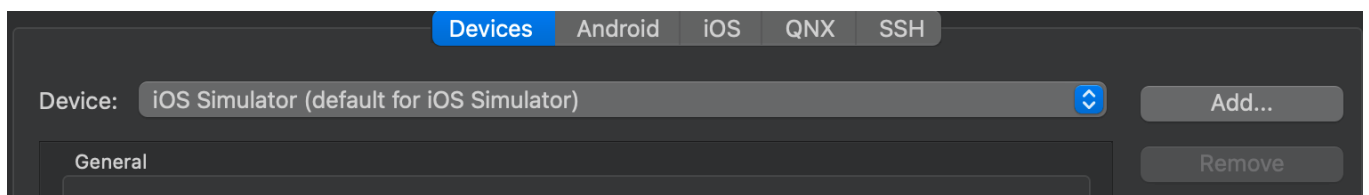
When you run the project, Qt Creator uses Xcode to deploy the application to the device.

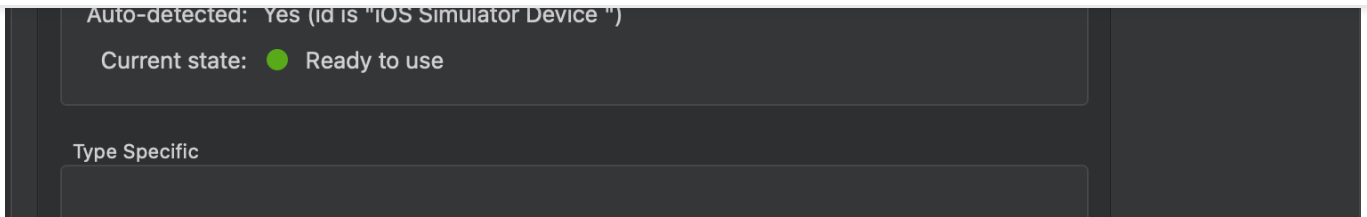
Your signing certificate is used to sign application packages for deployment to the device.

Note: If you cannot deploy applications because a provisioning profile is missing, check that provisioning profiles are listed in Xcode by selecting **Xcode > Preferences > Accounts**. For more information about how to acquire and install a provisioning profile, see Apple documentation.

Viewing Device Connection Status

When you connect an iOS device to your local machine with USB, Qt Creator automatically detects the device if you have configured it by using Xcode. To view information about the connected device, select **Preferences > Devices**.





If the current device state is **Connected**, (the traffic light icon is orange), you need to configure the device using Xcode.

Specifying Supported iOS Versions

You can build applications for the latest iOS version and deploy them to previous versions. For the most part, this works automatically. However, you must take care when you manually set your own target version. If you set it to a value higher than what Qt requires and supply your own file, you must add an entry to the that matches the value of `CMAKE_OSX_DEPLOYMENT_TARGET` (when using CMake), `QMAKE_IOS_DEPLOYMENT_TARGET` (when using qmake), or `cpp.minimumIosVersion` (when using Qbs) because iOS (and the App Store) will use the value as the authoritative one. `Info.plistLSMinimumSystemVersionInfo.plistLSMinimumSystemVersion`

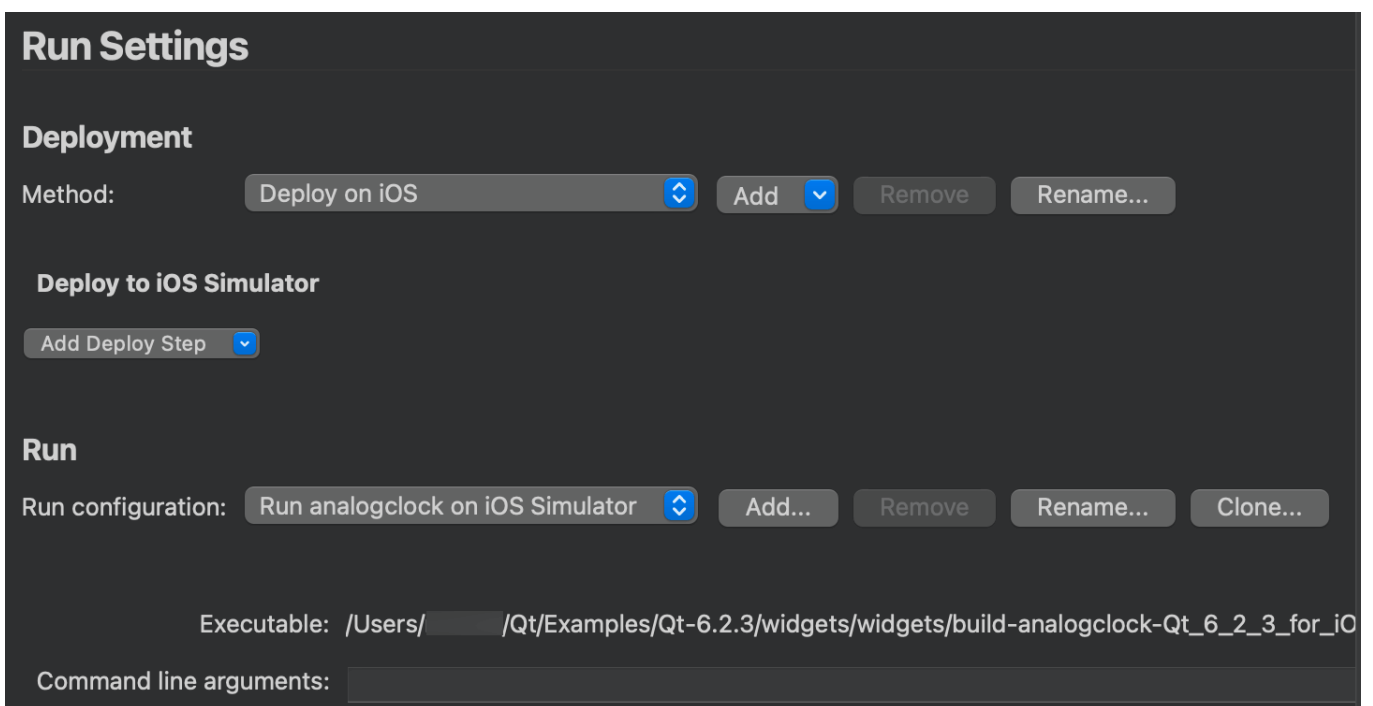
If you specify a deployment target value lower than what Qt requires, your application will almost certainly crash somewhere in the Qt libraries when run on an older version than Qt supports. Therefore, make sure that the actual build system code reflects the minimum iOS version that is actually required.

For more information, see [Expressing Supported iOS Versions](#).

Testing on Simulator

If you do not have an iOS device or you do not want to create an account, you can test applications on [Simulator](#), which is installed as part of Xcode. Each Xcode version simulates a predefined set of hardware devices and software versions.

You can change the simulated hardware and software version in the run settings for the project. Select **Projects > Run**, and then select the device to simulate in the **Device type** field.

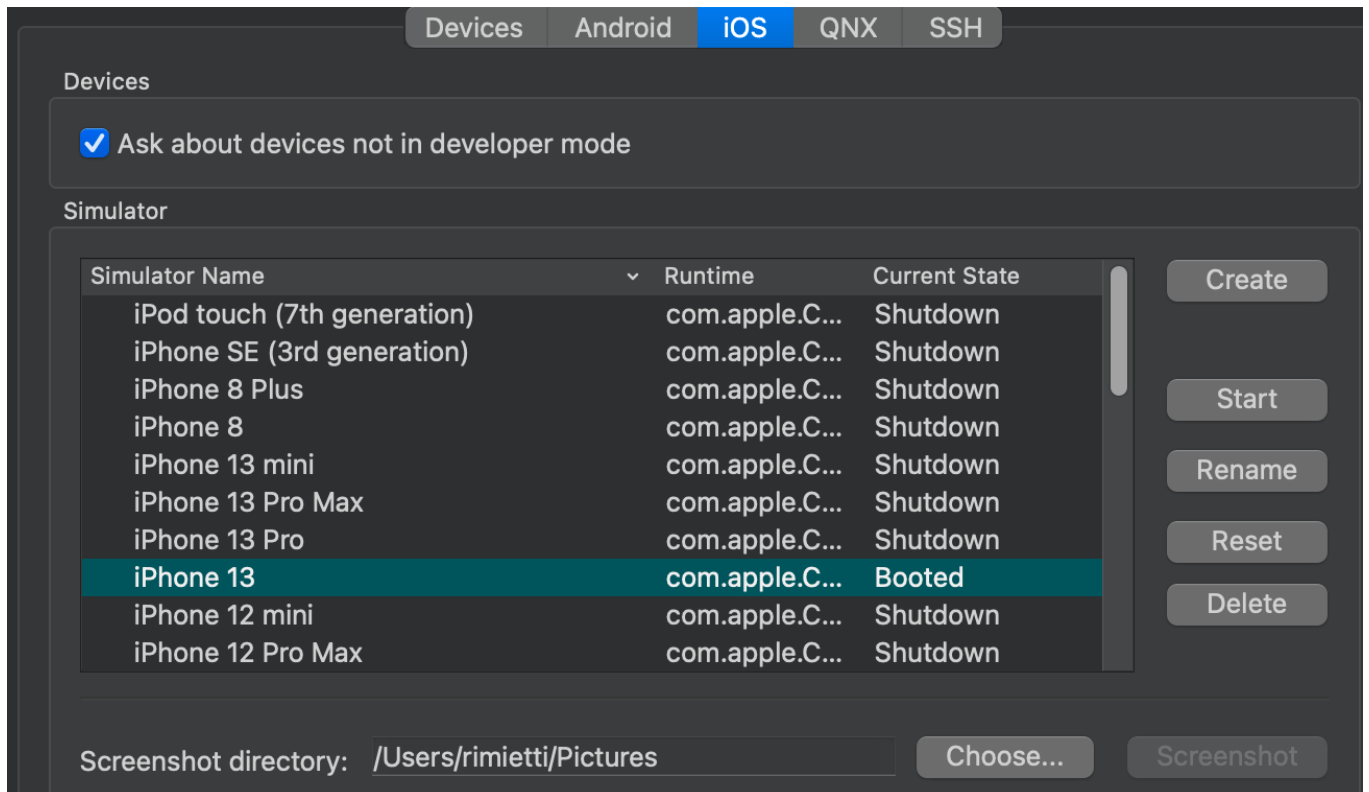


The simulator is started automatically when you run the application. To start the simulator manually, select **Preferences > Devices > iOS > Start**.

To take screenshots of the simulator, select **Preferences > Devices > iOS > Screenshot**. The screenshots are stored in the directory specified in the **Screenshot directory** field.

Managing Simulators

The available simulators are listed in **Preferences > Devices > iOS**.



To create a new simulator instance:

- Select **Create**.
- In the **Device type** field, select the device type from a list of devices supported by the Xcode version set as current on your local machine.
- In the **OS version** field, select an OS version from a list of OS versions supported by the selected device and the current Xcode version.

To rename the selected simulator, select **Rename**.

To reset the contents and settings of the selected simulators, select **Reset**.

To delete the selected simulator, select **Delete**.

Checking Current Xcode Version

To check the current Xcode version, enter the following command:

```
xcode-select --print-path
```

To change the version, enter the following command:

© 2022 The Qt Company Ltd. Documentation contributions included herein are the copyrights of their respective owners. The documentation provided herein is licensed under the terms of the [GNU Free Documentation License version 1.3](#) as published by the Free Software Foundation. Qt and respective logos are trademarks of The Qt Company Ltd in Finland and/or other countries worldwide. All other trademarks are property of their respective owners.



Contact Us

Company

- About Us
- Investors
- Newsroom
- Careers
- Office Locations

Support

- Support Services
- Professional Services
- Partners
- Training

Community

- Contribute to Qt
- Forum
- Wiki
- Downloads
- Marketplace

Licensing

- Terms & Conditions
- Open Source
- FAQ

For Customers

- Support Center
- Downloads
- Qt Login
- Contact Us
- Customer Success

