

How to Build Qt Application with PHYTEC Yocto BSPs

Step-by-step guide (Yocto)



The following example assumes you are using PD15.1.1 for the AM335x. Replace device specific names with the proper names for your device.

Make sure you have followed the Quickstart Guide and set up the build environment properly (see [Building the BSP from Source](#)). You will also need to have your hardware booted with your built images and accessible by ssh to deploy code from Qt.

1. To install the SDK for a Machine and Image type, you can create a SDK installer with bitbake in the BSP itself. Be sure that the correct target machine is set. You also need to pass the image type you want to create the sdk for:

```
bitbake -c populate_sdk phytec-qt5demo-image
```

This will take 1-5 hours depending on the image type and host machine. After that you may find the installer in your yocto directory under:

```
build/deploy/sdk
```

2. Install the SDK with (example):

```
user@ubuntu:/opt/PHYTEC_BSPs/yocto_ti/build/deploy/sdk$ ./yogurt-glibc-x86_64-phytec-qt5demo-image-cortexa8t2hf-vfp-neon-toolchain-AM335x-PD15.1.1.sh
```

3. Install qtcreator and source the installed SDK environment setup:

```
sudo apt-get install qtcreator
source /opt/yogurt/AM335x-PD15.1.1/environment-setup-cortexa8t2hf-vfp-neon-phytec-linux-gnueabi
qtcreator
```

4. In qtcreator, go to Tools -> Options -> Devices -> Add... -> Generic Linux Device and enter the required device information to connect to your board.
5. In the Options window, go to Build & Run -> Qt Versions -> Add... and browse to **/opt/yogurt/AM335x-PD15.1.1/sysroots/x86_64-yogurtsdk-linux/usr/bin/qt5/qmake**
6. Switch to the Compilers tab and add a new compiler. Browse to **/opt/yogurt/AM335x-PD15.1.1/sysroots/x86_64-yogurtsdk-linux/usr/bin/arm-phytec-linux-gnueabi/arm-phytec-linux-gnueabi-gcc**
7. Switch to the Debuggers tab and add a new debugger. Browse to **/opt/yogurt/AM335x-PD15.1.1/sysroots/x86_64-yogurtsdk-linux/usr/bin/arm-phytec-linux-gnueabi/arm-phytec-linux-gnueabi-gdb**
8. Click Apply. Switch to the Kits tab and enter a new kit with the device, compiler, debugger, and Qt versions you added in the last four steps. For **sysroot**, browse to **/opt/yogurt/AM335x-PD15.1.1/sysroots/cortexa8t2hf-vfp-neon-phytec-linux-gnueabi**
9. Click Apply to save your settings.
10. Create a new project (Other Project -> Empty Qt Project) and select the kit you created.
11. Add source files that you wish to compile.
12. Modify the .pro file to append the following lines of code (using HelloWorld project as an example, see Qt documentation for .pro file syntax):

```
QT += widgets
TARGET = HelloWorld
target.files = HelloWorld
target.path = /home/root/HelloWorld/
INSTALLS += target
```

13. Click Build -> Build All and save any changes. If there are no errors the build has succeeded.
14. Click Run to deploy your code to the remote device.

Related articles

Content by label

There is no content with the specified labels