



Y2 - Yocto Project Expert

Advanced Yocto Project usage and adaptation

Objectives

- Using and customizing Yocto
- Creating Yocto-based Embedded Linux platforms
- Using Yocto to develop components
- Building out of tree modules
- Setup Source cache

We use a recent Yocto version

Labs are conducted on qemu or on target boards, that can be:

- Dual Cortex/A7-based "STM32MP15-DISCO" boards from STMicroelectronics.
- Quad Cortex/A9-based "SabreLite" boards from NXP.
- Quad Cortex/A53-based "imx8q-evk" boards from NXP.

Prerequisite

- Good C programming skills
- Knowledge of Linux Embedded systems (see our [D1 - Embedded Linux with Buildroot and Yocto](#) course)
- Knowledge of Yocto Project Development (see our [Y1 - Yocto Project Development](#) course)
- Preferably knowledge of Linux user programming (see our [D0 - Linux user mode programming](#) course)

Course environment

- Printed course material (in English)
- One Linux PC for two trainees.
- One target platform for two trainees

Target Audience

- Any embedded systems engineer or technician with the above prerequisites.

Course Outline

First Day

Development process using the extensible SDK and devtool

- Using devtool to create a package and its recipe
- Using devtool to modify an existing package and recipe
- Using devtool to update a recipe to build a new version of a package

Exercise: Create, test and modify a recipe for an existing package using devtool

Develop and debug applications using SDK and eclipse

- Adding eclipse remote debug packages
- Configuring eclipse

Exercise: Create remote debugging session using eclipse

Writing tasks in python

- Introduction to python
- Using python in Yocto
 - The main bitbake classes
 - Defining variable values in Python
 - Writing tasks in Python

Exercise: Writing a task and customizing a recipe in Python

Porting Yocto

- Porting Yocto to a new board
- BSP architecture
 - Selecting and configuring u-boot recipe
 - Selecting and configuring kernel recipe
- Adding a new BSP layer (yocto-bsp create)

Exercise: Creating a new BSP layer

Second Day

BSP Development

- Adding a custom u-boot to Yocto
- Customizing the Yocto kernel recipe
 - Setting the default configuration
 - Adding patches
 - Specifying the kernel sources
- Configuring Linux Kernel
 - Using menuconfig
 - Using patches
 - Creating Configuration Fragments
 - Validating Configuration
- Kernel device tree

Exercise: Create u-boot and kernel recipes to use custom versions, test the result

Exercise: Patch kernel and activate new options using a fragment

Exercise: Create and use a new device tree

Out-of-Tree Modules

- Adding modules to image
- Creating an out-of-tree module
- Kernel modules with eSDK

Exercise: Build and test modules

Tailoring the build system

- Setting up a Yocto source cache
 - Local, per system, cache setup
 - Setting up a global, network wide, cache
- Customizing the build system
 - Using a prebuilt toolchain
 - Using a pre-compiled kernel
- Optimizing Yocto build times
 - Using prebuilt, binary, packages
 - Using shared compilation caches

Exercise: Setting up a global source cache

Exercise: Setting up an optimized build environment and rebuilding an image