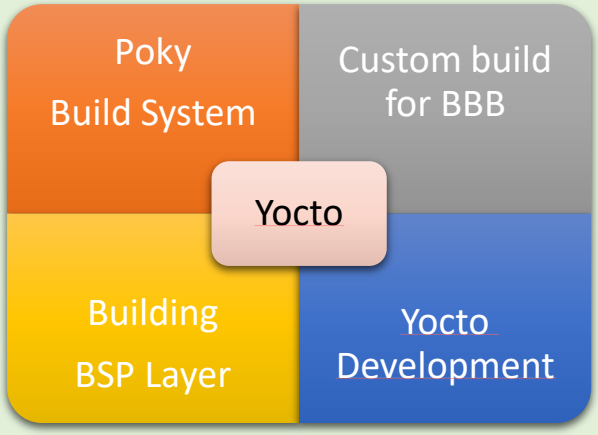
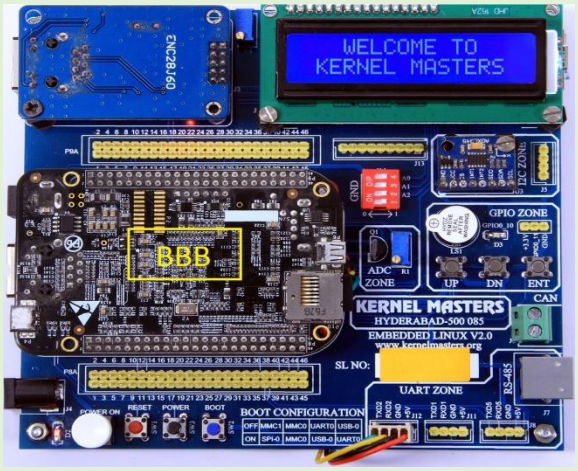
KERNEL MASTERS

Yocto Project with Beaglebone Black





Who should attend?

- Engineers who are migrating to a yocto Project based distribution environment
- Project managers and engineers who are evaluating the use of Linux for their projects
- Project managers and engineers who are considering supporting their own Linux distribution and tools.

Hardware used

- The practical exercises will be run on a Beagle Bone Black (BBB) with a Cortex ARM.
- All exercises will be applicable to any other type of board supported by Linux.
- Kernel Masters Designed Beagle bone expansion board. For purchase this board contact Kernel Masters Team.

Prerequisites:

- Delegates must have attended Developing with Embedded Linux or an equivalent course, or have some experience of using Linux for Embedded Systems.
- We assume that attendees are fully fluent in C, data structures and that the Linux/Unix command line is a familiar environment.
- Linux System Programming & Device Drivers.
- Embedded Linux Porting knowledge.

Course Material:

https://github.com/kernelmasters

Note: All Sessions are highly interactive hands-on-sessions with KM-BBB

KERNEL MASTERS

Yocto with Beaglebone black Detailed Agenda

Yocto with Beaglebone Black Summary: (Detailed agenda below)

Part 1: Introduction
Part 2: Yocto Project Overview and Concepts
Part 3: Yocto Project Customization
Part 4: Yocto Build – lab Sessions

Yocto with Beaglebone Black Detailed Syllabus:

Part 1: Introduction

Legacy Embedded Linux

- Embedded Linux Boot sequence
- u-boot customization
- kernel customization
- rootfs customization

Legacy Embedded Linux vs Yocto Embedded Linux
Yocto versions vs Linux kernel versions vs u-boot versions
Yocto Project Documentation

Part 2: Yocto Project Overview and Concepts

Session 1: Introducing the Yocto Project

1.1. What is the Yocto Project?
  1.1.1. Features
  1.1.2. Challenges
1.2. Why Yocto?
1.3. Yocto Project Source tree
1.4. The Yocto Project Layer Model
1.5. Components and Tools
1.6. Development Tools
1.7. Production Tools
1.8. Open-Embedded Build System Components
1.9. Reference Distribution (Poky)
1.10. Development Methods
1.11. Reference Embedded Distribution (Poky)
1.12. The OpenEmbedded Build System Workflow

Session 2: The Yocto Project Development Environment

2.1. Open Source Philosophy
2.2. The Development Host
2.3. Yocto Project Source Repositories
2.4. Git Workflows and the Yocto Project
2.5. Git
2.6. Licensing

Session 3: Yocto Project Concepts

3.1. Yocto Project Components
  3.1.1. BitBake
  3.1.2. Recipes
  3.1.3. Classes
  3.1.4. Configurations
3.2. Layers
3.3. OpenEmbedded Build System Concepts
  3.3.1. User Configuration
  3.3.2. Metadata, Machine Configuration, and Policy Configuration

3.3.3. Sources
3.3.4. Package Feeds
3.3.5. BitBake
3.3.6. Images
3.3.7. Application Development SDK
3.4. Cross-Development Toolchain Generation
3.5. Shared State Cache
3.5.1. Overall Architecture
3.5.2. Checksums (Signatures)

Part 3: Yocto Project Customization

Session 1: Setting Up to Use the Yocto Project

1.1. Creating a Team Development Environment
1.2. Preparing the Build Host
1.3. Locating Yocto Project Source Files
1.4. Cloning and Checking Out Branches

Session 2: Common Tasks

2.1. Understanding and Creating Layers
2.2. Writing a New Recipe
2.3. Adding a New Machine
2.4. Building
2.5. Speeding Up a Build
2.6. Working With Libraries
2.7. Using x32 psABI
2.8. Enabling GObject Introspection Support
2.9. Optionally Using an External Toolchain
2.10. Creating Partitioned Images Using Wic
2.11. Flashing Images Using bmaptool
2.12. Making Images More Secure
2.13. Creating Your Own Distribution
2.14. Creating a Custom Template Configuration Directory
2.15. Conserving Disk Space During Builds
2.16. Working with Packages

Session 3: Using the Quick EMUlator (QEMU)

3.1. Overview
3.2. Running QEMU
3.3. Switching Between Consoles
3.4. Removing the Splash Screen
3.7. QEMU CPU Compatibility Under KVM
3.8. QEMU Performance
3.9. QEMU Command-Line Syntax
3.10. runqemu Command-Line Options

Part 4: Lab Sessions

1. Yocto Building image on x86
2. Yocto Building image on BeagleBone Black [BBB]
3. Yocto Building image on Kernel Masters Beaglebone black expansion Board.

Authored and Compiled By: Boddu Kishore Kumar
Email: kishore@kernelmasters.org
Reach us online: www.kernelmasters.org
Contact: 9949062828