



Linux Yocto System Engineer Syllabus



1-Month Course (Fast-Track)

Week 1 – Intro & Setup

- ❖ Embedded Linux vs Desktop Linux
- ❖ Yocto Project overview (Poky, BitBake, Layers)
- ❖ Host setup and first Yocto build (core-image-minimal)
- ❖ Running images on QEMU

Week 2 – Recipes & Layers

- ❖ Anatomy of a recipe (.bb, .bbappend)
- ❖ Adding/removing packages in an image
- ❖ Creating a custom layer (meta-training)
- ❖ Writing a simple recipe (C “Hello World” app)

Week 3 – Kernel & BSP

- ❖ Kernel basics (menuconfig, patches, modules)
- ❖ BSP concepts (meta-raspberrypi, meta-ti)
- ❖ Flashing image to Raspberry Pi 5 / BeagleBone
- ❖ Boot logs analysis

Week 4 – Custom Images & SDK

- ❖ Creating custom image (training-image)
- ❖ Adding system utilities (SSH, nano, htop)
- ❖ Generating SDK and cross-compiling apps
- ❖ Capstone mini-project: Custom Yocto image with one application

3-Month Yocto Professional Course

Week 1 – Embedded Linux & Yocto Basics

- ❖ **Class 1 (Theory):** Embedded Linux vs Desktop, Toolchains, Boot Process
- ❖ **Class 2 (Lab):** Linux basics refresher + host PC setup (Ubuntu, Git, GCC, packages)
- ❖ **Class 3 (Theory):** Yocto Project overview (Poky, BitBake, Layers, Recipes)
- ❖ **Class 4 (Lab):** First Yocto build (core-image-minimal) + run in QEMU

Week 2 – Yocto Structure & Configuration

- ❖ **Class 5 (Theory):** Poky directory structure, BitBake workflow
- ❖ **Class 6 (Lab):** Exploring local.conf and bblayers.conf
- ❖ **Class 7 (Theory):** Metadata, recipes, tasks, variables
- ❖ **Class 8 (Lab):** Modify build to include nano / htop in image

Week 3 – Recipes Deep Dive

- ❖ **Class 9 (Theory):** Anatomy of a recipe (.bb, .bbappend, .bbclass)
- ❖ **Class 10 (Lab):** Writing a recipe for C “Hello World” app
- ❖ **Class 11 (Lab):** Recipe fetching from GitHub + packaging
- ❖ **Class 12 (Theory + Demo):** Patching with Quilt

Week 4 – Layers & Customization

- ❖ **Class 13 (Theory):** Layer concepts, compatibility, priority
- ❖ **Class 14 (Lab):** Creating custom layer (meta-training)
- ❖ **Class 15 (Lab):** Using .bbappend to modify BusyBox
- ❖ **Class 16 (Mini-Project):** Add a custom shell script recipe to image

Week 5 – Kernel Basics

- ❖ **Class 17 (Theory):** Kernel recipes, config, patches
- ❖ **Class 18 (Lab):** Kernel build with menuconfig
- ❖ **Class 19 (Lab):** Enable I2C/SPI in kernel config
- ❖ **Class 20 (Lab):** Build and deploy custom kernel on Raspberry Pi 5

Week 6 – Kernel Modules & Device Drivers

- ❖ **Class 21 (Theory):** Kernel modules in Yocto
- ❖ **Class 22 (Lab):** Write a simple LED blink kernel module
- ❖ **Class 23 (Lab):** Package kernel module with Yocto recipe
- ❖ **Class 24 (Demo):** Device Tree basics

Week 7 – Bootloader (U-Boot)

- ❖ **Class 25 (Theory):** Boot sequence (ROM → Bootloader → Kernel → RootFS)
- ❖ **Class 26 (Lab):** Build and configure U-Boot for Raspberry Pi 5
- ❖ **Class 27 (Lab):** Modify U-Boot environment variables
- ❖ **Class 28 (Lab):** Debug boot logs via UART

Week 8 – BSP Development

- ❖ **Class 29 (Theory):** BSP structure, meta-raspberrypi overview
- ❖ **Class 30 (Lab):** Add support for external I2C device
- ❖ **Class 31 (Lab):** Modify Device Tree overlay and test on hardware
- ❖ **Class 32 (Mini-Project):** Create a minimal BSP customization for Pi 5

Week 9 – Custom Images & SDK

- ❖ **Class 33 (Theory):** Image types (core-image-base, full-cmdline)
- ❖ **Class 34 (Lab):** Create custom image (training-image)
- ❖ **Class 35 (Lab):** Boot Pi 5 with custom image (with SSH, htop, nano)
- ❖ **Class 36 (Lab):** Generate SDK and use it to build a simple app outside Yocto

Week 10 – Application Packaging

- ❖ **Class 37 (Theory):** Application development workflow in Yocto
- ❖ **Class 38 (Lab):** Package Python GPIO blink script
- ❖ **Class 39 (Lab):** Add C++ app recipe using SDK
- ❖ **Class 40 (Lab):** Auto-start service using systemd

Week 11 – Debugging & Profiling

- ❖ **Class 41 (Theory):** Debugging tools (gdb, strace, logs)
- ❖ **Class 42 (Lab):** Debugging app with gdbserver on Pi 5
- ❖ **Class 43 (Lab):** Profiling with perf
- ❖ **Class 44 (Demo):** Optimize image size by stripping debug packages

Week 12 – Capstone Project & Wrap-Up

- ❖ **Class 45 (Kickoff):** Define project (custom Yocto distro + app + BSP mods)
- ❖ **Class 46 (Lab):** Build & integrate custom kernel/app into image
- ❖ **Class 47 (Lab):** Flash to Pi 5 & live demo
- ❖ **Class 48 (Presentations + Review):** Student project presentations + Final Q&A

6-Month Yocto Comprehensive Course

Week 1 – Introduction to Embedded Linux

- ❖ **Class 1:** Embedded systems vs Desktop Linux
- ❖ **Class 2:** Linux architecture, kernel, rootfs, toolchain
- ❖ **Class 3:** Boot process in embedded Linux
- ❖ **Class 4:** Lab – Setting up Ubuntu host & essentials

Week 2 – Yocto Basics

- ❖ Class 5: Yocto Project overview (Poky, BitBake, Layers, Metadata)
- ❖ Class 6: Directory structure of Poky
- ❖ Class 7: First build with core-image-minimal
- ❖ Class 8: Lab – Running images in QEMU

Week 3 – BitBake & Configurations

- ❖ Class 9: BitBake tasks, recipes, dependencies
- ❖ Class 10: Configuration files (local.conf, bblayers.conf)
- ❖ Class 11: Build customization (adding packages)
- ❖ Class 12: Lab – Add nano & htop into image

Week 4 – Recipes Basics

- ❖ Class 13: Anatomy of a recipe (.bb, .bbclass, .bbappend)
- ❖ Class 14: Creating a custom recipe (C Hello World app)
- ❖ Class 15: Fetching sources from GitHub/tarballs
- ❖ Class 16: Lab – Patch management with Quilt

Week 5 – Layers & Appends

- ❖ Class 17: Layer concept and priority
- ❖ Class 18: Creating a new custom layer (meta-student)
- ❖ Class 19: Using .bbappend to modify recipes
- ❖ Class 20: Lab – Modify BusyBox via append

Week 6 – Advanced Recipes

- ❖ Class 21: Inheritance & classes in recipes
- ❖ Class 22: Packaging & splitting output packages
- ❖ Class 23: Lab – Add Python script as Yocto recipe
- ❖ Class 24: Lab – Recipe with external dependencies

Week 7 – Custom Images

- ❖ Class 25: Image types (core-image-base, full-cmdline)
- ❖ Class 26: IMAGE_FEATURES & DISTRO_FEATURES
- ❖ Class 27: Lab – Create training-image with utilities
- ❖ Class 28: Boot Pi 5/BeagleBone with custom image

Week 8 – SDK & Devtool

- ❖ Class 29: Yocto SDK & Ext-SDK basics
- ❖ Class 30: Lab – Build SDK and install
- ❖ Class 31: Using SDK for external app builds
- ❖ Class 32: Lab – Using devtool to modify BusyBox

Week 9 – Kernel Basics

- ❖ Class 33: Linux kernel in Yocto
- ❖ Class 34: Configuring kernel with menuconfig
- ❖ Class 35: Lab – Build custom kernel with I2C enabled
- ❖ Class 36: Deploy kernel to board

Week 10 – Kernel Modules

- ❖ Class 37: Writing kernel modules (LED blink driver)
- ❖ Class 38: Lab – Build module outside Yocto (DKMS style)
- ❖ Class 39: Recipe for kernel module in Yocto
- ❖ Class 40: Lab – Test kernel module on hardware

Week 11 – Bootloader (U-Boot)

- ❖ Class 41: Boot sequence explained
- ❖ Class 42: U-Boot in Yocto
- ❖ Class 43: Lab – Modify U-Boot environment variables
- ❖ Class 44: Debug boot logs with UART

Week 12 – Device Tree

- ❖ Class 45: Device Tree basics
- ❖ Class 46: Device Tree overlays in Raspberry Pi/BeagleBone
- ❖ Class 47: Lab – Enable UART/I2C in device tree
- ❖ Class 48: Lab – Rebuild DTB & deploy

Week 13 – BSP Concepts

- ❖ Class 49: BSP structure & meta-layers (meta-raspberrypi, meta-ti)
- ❖ Class 50: Porting BSP basics
- ❖ Class 51: Lab – Explore meta-raspberrypi BSP
- ❖ Class 52: Lab – Build & test BSP image

Week 14 – BSP Customization

- ❖ Class 53: Modify BSP configs
- ❖ Class 54: Add support for external I2C/SPI device
- ❖ Class 55: Lab – Overlay for sensor
- ❖ Class 56: Lab – Debugging BSP boot issues

Week 15 – Advanced BSP Work

- ❖ Class 57: Machine configs & overrides
- ❖ Class 58: Adding custom board definitions
- ❖ Class 59: Lab – Custom BSP for student board
- ❖ Class 60: Presentations of BSP modifications

Week 16 – Mid-Term Project

- ❖ Class 61: Project kickoff – Custom BSP + Image
- ❖ Class 62: Hands-on session
- ❖ Class 63: Troubleshooting session
- ❖ Class 64: Mid-term project review

Week 17 – Application Packaging

- ❖ Class 65: Packaging C/C++ apps
- ❖ Class 66: Lab – Add networking utility app
- ❖ Class 67: Packaging Python apps
- ❖ Class 68: Lab – Flask web server recipe

Week 18 – System Services

- ❖ Class 69: Init vs systemd in Yocto
- ❖ Class 70: Lab – Add systemd service for Python app
- ❖ Class 71: Auto-start apps in Yocto image
- ❖ Class 72: Lab – Debug systemd logs

Week 19 – Debugging Tools

- ❖ Class 73: Debugging apps with gdb/gdbserver
- ❖ Class 74: Using strace for system calls
- ❖ Class 75: Lab – Perf profiling
- ❖ Class 76: Lab – Ftrace basics

Week 20 – Optimization

- ❖ Class 77: Image size optimization
- ❖ Class 78: Using PACKAGECONFIG
- ❖ Class 79: Lab – Optimize BusyBox build
- ❖ Class 80: Lab – Compare image sizes

Week 21 – Advanced Yocto

- ❖ Class 81: BitBake internals (events, tasks)
- ❖ Class 82: Multiconfig builds
- ❖ Class 83: Lab – Multi-machine build
- ❖ Class 84: Lab – Advanced overrides

Week 22 – Security

- ❖ Class 85: RootFS hardening (remove services, add users)
- ❖ Class 86: Kernel hardening basics
- ❖ Class 87: Lab – Secure image with SSH keys
- ❖ Class 88: License compliance (SPDX, Yocto tools)

Week 23 – Capstone Project Work

- ❖ Class 89: Define final project (custom Yocto distro)
- ❖ Class 90: Hands-on build sessions
- ❖ Class 91: Debugging sessions
- ❖ Class 92: Testing project on hardware

Week 24 – Final Wrap-Up

- ❖ Class 93: Capstone polishing
- ❖ Class 94: Student project presentations
- ❖ Class 95: Review & advanced career guidance
- ❖ Class 96: Final Q&A + Certification ceremony