

# EMBEDDED SYSTEM

## Course Outlines:

### Introduction to the GNU/Linux Operating Systems

Why Linux?

Embedded Linux Today

Open Source and the GPL

Standards and Relevant Bodies

### The Big Picture

Embedded or Not?

Anatomy of an Embedded System

Storage Considerations

Embedded Linux Distributions

### Processor Basics

Stand-Alone Processors

Integrated Processors: Systems on Chip

Other Architectures

Hardware Platforms

### The Linux Kernel: A Different Perspective

Linux Kernel Construction

Kernel Configuration

Obtaining a Custom Linux Kernel

### Kernel Initialization

Composite Kernel Image: Piggy and Friends

Initialization Flow of Control

Kernel Command-Line Processing

Subsystem Initialization

The init Thread

### User Space Initialization

Root File System

Kernel's Last Boot Steps

The init Process

Initial RAM Disk

### Bootloaders

Role of a Bootloader

Bootloader Challenges

A Universal Bootloader: Das U-Boot

Porting U-Boot

Device Tree Blob (Flat Device Tree)

Other Bootloaders

## **Device Driver Basics**

- Device Driver Concepts
- Loadable Modules
- Device Driver Architecture
- Minimal Device Driver Example
- Module Build Infrastructure
- Installing a Device Driver
- Module Parameters
- Module Utilities
- Driver Methods
- Driver File System Operations
- Building Out-of-Tree Drivers
- Device Drivers and the GPL

## **File Systems**

- Linux File System Concepts
- Partitions
- Mounting a File System
- Checking File System Integrity
- Building a JFFS2 Image
- Network File System
- Root File System on NFS
- Pseudo File Systems
- Other File Systems
- Building a Simple File System

## **MTD Subsystem**

- MTD Overview
- Enabling MTD Services
- MTD Basics
- Configuring MTD on Your Target
- MTD Partitions
- Redboot Partition Table Partitioning
- Kernel Command-Line Partitioning
- Mapping Driver
- Flash Chip Drivers
- Board-Specific Initialization
- MTD Utilities
- JFFS2 Root File System
- UBI File System

## **Introduction to BusyBox**

- BusyBox Is Easy
- BusyBox Configuration
- Cross-Compiling BusyBox
- BusyBox Operation
- BusyBox init
- BusyBox Target Installation
- BusyBox Applets

## **Host System Requirements**

- Embedded Development Environment
- Cross-Development Environment
- Hardware Debug Probe
- Hosting Target Boards
- TFTP Server
- BOOTP/DHCP Server
- NFS Server
- Development Tools
- GNU Debugger (GDB)
- Debugging a Core Dump
- Invoking GDB
- Debug Session in GDB
- Data Display Debugger
- Tracing and Profiling Tools
- Binary Utilities
- Miscellaneous Binary Utilities

## **Kernel Debugging**

- Techniques Challenges to Kernel Debugging
- Using KGDB for Kernel Debugging
- KGDB Kernel Configuration
- Target Boot with KGDB Support
- Useful Kernel Breakpoint
- Kernel Debugging Techniques
- Debugging Optimized Kernel Code
- GDB User-Defined Commands
- Debugging Loadable Modules
- Hardware-Assisted Debugging
- Programming Flash Using a JTAG Probe
- Debugging with a JTAG Probe

## **Debugging Embedded Linux Applications**

- Target Debugging Remote (Cross) Debugging
- Debugging with Shared Libraries Shared Library Events in GDB
- Additional Remote Debug Options
- Debugging Using a Serial Port Attaching to a Running Process

## **Open Source Build Systems**

- Why Use a Build System?
- Scratchbox
- Installing Scratchbox
- Creating a Cross-Compilation Target
- Buildroot
- OpenEmbedded
- Building Images

## **Linux and Real Time**

What Is Real Time?

Soft Real Time

Hard Real Time

Linux Scheduling

Kernel Preemption

Impediments to Preemption

Preemption Models SMP Kernel

Sources of Preemption

Latency Real-Time Kernel Patch

Real-Time Features

Creating a Real-Time Process

Real-Time Kernel Performance Analysis

Soft Lockup Detection

## **Universal Serial Bus**

USB Overview

USB Topology

USB Connectors

USB Cable Assemblies

USB Modes

Configuring USB

Useful USB Tools

USB File System

Useful USB Miscellaneous

## **udev**

What Is udev?

Device Discovery

Default udev Behavior

Understanding udev Rules

Modalias

Typical udev Rules Configuration

Initial System Setup for udev

Loading Platform Device Drivers

Customizing udev Behavior

Persistent Device Naming

udev Helper Utilities

Using udev with busybox