Programming Embedded Linux

Course 105 – 40 Hours

Overview

This course is intended for embedded systems developers taking their first steps with Linux as an embedded system platform, and provides them with the skills required for becoming a productive programmer in that environment. The curriculum includes building applications and device drivers with real time constraints and understanding the inner workings of the Linux system and its effects on system behavior.

The course is suitable for people planning to develop for embedded Linux platforms from any source, including "home made" Linux distributions or embedded Linux system vendors.

Course Objectives

Create applications and device drivers for Embedded Linux environments, or to import such applications from systems using legacy RTOS.

Who Should Attend

Seasoned embedded systems developers wishing to become competent Linux Embedded systems developers.

Prerequisites

Students should have a working knowledge with C programming language and basic knowledge with embedded systems.

Course Contents

Introduction

- What is Linux
- Layers in a Linux system
- Linux vs. Legacy RTOS

Basic concepts

- Files and file system
- The shell
- Basic commands
- Processes
- Setting up networking

Application programming and the user space API

- Makefiles and the build environment
- Processes and threads
- Real time priorities

info@mabel-tech.com

52 Bar Yehuda St. - Nesher
Phone: +972-4-8571119
Fax: +972-4-8570307

דרך בר יהודה 52, נשר
טלפון: 04-8571119
פקס: 04-8570307

- Synchronization and IPC (mutex, condition variables, mailboxes, pipes, shared memory, Unix domain sockets and signals)
- Timers
- Memory mapping and locking
- Debugging applications: in process and using remote debugger
- Labs – using pipes, debugging

Linux Kernel
- Kernel overview
- History
- Versions
- Source code layout
- Good practices
- System call interface

Writing a simple kernel module
- A simple kernel module structure
- Implicit steps of compiling modules in Linux kernel version 2.6
- Using shell commands to manipulate modules
- The kernel log
- Using the printk function
- Passing parameters to the module

Memory Management
- Memory areas
- Memory page frames
- Requesting and releasing page frames
- Allocating contiguous virtual memory area
- The slab and slob allocators
- Memory caches and allocations
- Managing slabs
- Creating and destroying caches
- User space memory access

Implementing a character device file
- The VFS structure
- Initialization and termination
- Opening the device file
- IOCTL
- Implementing base operations

Debugging
- Kernel configuration for debugging
- printk
- KGDB
- Oops messages

Locking mechanisms
- Locking requirements
- Preemption
- Atomic bit operations
- Interrupt disabling
- Spin lock
- Semaphores

Linux Scheduler
- Process and thread
- Scheduling policies
- Priorities
- Kernel tasks
- task_struct structure
- SMP scheduling

Interrupt handling
- Hardware interrupt handling basics
- Interrupt handler and control
- Low level handling
- Wait queues technique
- Threaded interrupts

Bottom halves
- Differing work
- Using software interrupts
- Tasklets
- Timers & RTC
- Work queues

Network sub system overview
- The layer model
- Registration and un-registration
- Socket buffers, allocations and manipulations
- Network headers
- Packet reception
- Packet transmission
- NAPI

info@mabel-tech.com

52 Bar Yehuda St. - Nesher
Phone: +972-4-8571119
Fax: +972-4-8570307

דרך בר יהודה 52, נשר
טלפון: 04-8571119
פקס: 04-8570307