ADAPTIVE INFOTECH
New No.304, Old No.88
East Perumal Maistry Street
Madurai 625 001. Tamil Nadu. India.
www.adaptit.co.in Ph.: +(91)452-2330521

Embedded Linux

June 2025

Course Outline

- Introduction to Linux and Embedded Systems
  - A few examples of embedded systems running Linux
  - Advantages of Linux versus traditional embedded operating systems
  - Embedded Linux Development Environment
  - Essential Linux Commands
- Real Time Operating System
  - Concurrency and Parallelism
  - Multithreading concepts
  - Thread synchronisation techniques
  - Thread Management
  - Process management
  - Process Scheduling
  - Inter-Process Communication (IPC)
- Cross-compilation Toolchain and C library
  - Definition and Components
  - Toolchain Options
  - Obtaining a Toolchain
  - Cross-compiling user-space libraries and applications
- Practical view of the Linux System
- Hardware Architecture
- Embedded-Linux Operating System Architecture
- Linux-based Embedded System Component Stack
- Anatomy of a Linux-based System
- Open-source Components
- Open-source Licenses and compliance
- Introduction to Buildroot
- System Configuration, Build Process
  - Embedded system building tools
- Boot Process
  - Introduction
  - Booting on Embedded Systems
  - Bootloaders
  - The U-Boot Bootloader
  - Trusted Firmware
  - Boot Process on ARM platforms
- Linux Kernel Modules and Programming

- Linux kernel sources
- Kernel configuration
- Compiling and installing the kernel
- Booting the kernel
- Kernel C Programming
- Make Utility

- Accessing hardware devices
  - Kernel drivers
  - User-space interface to drivers
  - Communication between Kernel Space and User Space
  - Using kernel modules
  - Hardware access using /dev and /sys
  - Describing non-discoverable hardware – Device Tree
  - Discoverable hardware – USB and PCI
- C library
- System Call Overview
- Linux Root File System
  - Principle and solutions
  - Contents
  - Pseudo filesystems
  - Minimal filesystem
  - Device Files
  - Virtual Filesystems
- Block File System
  - Accessing and partitioning block devices
  - Read-only block systems
  - RAM filesystems
- Flash storage and filesystems
  - The Memory Technology Devices (MTD) filesystem
  - Kernel configuration options
  - MTD storage partitions
- Memory Management
  - Virtual memory and paging
  - Memory allocation and deallocation
  - Memory mapping and system calls
- Memory-centric System Model
- Persistent storage
- BusyBox
    - Features
    - Configuration, Compiling, Deploying
- Linux Directory Structure
  - Understanding the root directory and subdirectories

- File paths and symbolic links
- Input and output
  - C System Calls
  - File Management
- Networking
  - Networking basics
  - Socket Programming
  - Serial Port Programming
- Embedded Linux Application Development
  - System debugging, remote debugging
  - Tracing and profiling
  - Memory debugging
- Advanced Topics
  - Third party libraries and applications
- Embedded C Programming
  - APIs and Libraries for Peripherals and Interfaces
- Mini Project
  - Hands-on project involving the embedded Linux
  - File storage on SD cards and usage of software repositories
  - Bluetooth Low Energy communication with a smartphone
  - Embedded web server over a wi-fi connection
  - Designing and implementing an embedded system project

Benefits of the course

- R&D job opportunities in core areas of communication, automation, medical, automotive, etc.
- High-end scalable and software defined product development
- IoT/AIoT/CIoT/IIoT/MIoT/xIoT devices development
- Advanced Robotics development
- Audio, Video, Networking applications
- Learn new concepts from the industry experts
- Develop job relevant skills with hands-on projects
- Excellent platform for developing innovative AI&ML products
- Course completion certificate

Students will learn

- Basics of Embedded Linux
- ARM architecture and programming

- Installation and usage of tool chain, Keil, CCS
- Configuration via Device Tree Specification
- Booting Process
- Root File System,
- Modularity and Reutilisation Principles - Functions
- Inter Process Communication
- Memory Management
- Board Support Package
- Device Driver Development
- Peripherals and Interfaces
- Explain the fundamentals of how embedded Linux based systems are developed
- Implement and debug ARM based embedded system employing RT-Linux
- Debug terminal for code / scripts development and analysing log reports
- Develop maintainable software
- Design and develop mini project - audio message playback and video applications
- Report date and time of alarms to PC based on the RTC
- Describe how the concept of modularity can be applied to embedded systems programming for maintainability and increasing productivity especially in a complex environment / application development by refactoring the program code
- Describe how to connect an SD card to embedded system board using an SPI bus interface
- Develop programs to manage files on an SD card with the embedded system board
- Implement a revision control system using repositories
- Summarise the fundamentals of file systems
- Describe how to connect Bluetooth Low Energy (BLE) modules to the embedded system board using UART
- Develop programs to exchange data between the embedded system board and a smartphone using BLE connection
- Describe how to connect a Wi-Fi module to the embedded system board
- Develop programs to serve a web page using the embedded system board and a Wi-Fi module
- Summarise the fundamentals of AT commands and TCP/IP connections
- Describe how an embedded system project can be developed following an ordered process
- Design and implement a prototype of an embedded system, including its hardware and software
- Summarise the fundamentals of the concepts of verification and validation
- Fundamentals of entire life cycle of any embedded linux based advanced product design
- Integrating IMU to portable IoT devices with positioning capabilities
- Embedded C programming
- Power Saving Modes

- Hardware Accelerators
- Communication Ports, Ethernet, BLE, WiFi, TCP/IP
- Controlling Graphical LCD Display
- DC Motor Driving using Relays and Interrupts
- Software execution time optimisation using neon programming
- Socket programming, Cloud Access, IoT Products Design
- Case studies: smart street lighting, implementation of PWM to control brightness of an RGB LED
- Indoor environment monitoring
- Use of debug and event log messages, developing text/graphical user interface code
- .json file, Hardware Abstraction Layer
- Keil Studio Cloud
- Time Management
- Fundamentals of Linux System Programming, including Processes and Threads
- How to use Buildroot to build a Custome Linux Kernel and root filesystem for an embedded device
- Skills that you gain are System Programming, Build Tools, OS Process Management, Cross Platform Development, Operating Systems, Shell Script, Debugging, C and C++, Linux, Embedded Systems, Filesystems.
- Understand the overall architecture of Embedded Linux Systems.
- Choose, build, setup and use a cross-compilation toolchain.
- Understand the booting sequence of an Embedded Linux System, and to set up and use the U-boot boot loader.
- Select a Linux kernel version, to configure, build and install the Linux kernel on an embedded system
- Create from scratch a Linux root filesystem, including all its elements: directories, applications, configuration files, libraries.
- Choose and set up the main Linux filesystem for block and flash storage devices, and understand their main characteristics
- Interact with the hardware devices, configure the kernel with appropriate drivers and extend the Device Tree.
- Select, cross-compile and integrate open-source software components (libraries, applications) in an Embedded Linux System, and to handle license compliance.
- Setup and use an embedded Linux build system, to build a complete system for an embedded platform.
- Develop and debug applications on an embedded Linux system.

Instructor: Rishikeshlalbabu S Ramiya

https://www.linkedin.com/in/rishikeshlalbabu

ADAPTIVE INFOTECH
New No.304, Old No.88
East Perumal Maistry Street
Madurai 625 001. Tamil Nadu. India.
www.adaptit.co.in Ph.: +(91)452-2330521

Instructor's Training Expertise

- On behalf of AdaptIT, signed an MoU with Dhanalakshmi College of Engineering, Chennai to jointly deliver various technical programmes such as training, final year projects guidance, etc.

- Conducted the following Corporate Trainings
  - CDMA (2G and 3G) – Wipro, Bangalore, ElementK(NIIT), Chennai.
  - GSM – ElementK (NIIT), Chennai.
  - Bluetooth (Protocol Layers, Profiles, IOT, Qualification Process) – L&T, Bangalore, Atheros, Chennai.
  - Network Management – Siemens, Bangalore.
  - Telecom & Datacom – Alcatel-Lucent, Chennai.
  - Wireless Communication Implementation Aspects – Tata Elxsi, Chennai, CDAC, Trivandrum
  - High Speed Transceivers (DDR3, CPRI, SRIO, Gigabit Ethernet) – Altera, Malaysia
  - Image and Video Processing – Altera, Malaysia
  - CPRI Version 4.2 – Tata Elxsi, Bangalore
  - VoIP & SIP – MIMOS, Malaysia, Sapura Secure Technologies, Malaysia
  - Linux – American Mega Trends, Chennai.

- Conducted the following Student Trainings
  - Embedded systems
  - Signals and Systems
  - CDMA / WCDMA
  - GSM / GPRS
  - C/C++ Language
  - DTP/Basic Computer Knowledge

Instructor's Industrial Experience

- Developed the following AdaptIT products
  - Stand-alone GPS product Hardware and Software for Position Data Logging and Speed Sensing Applications
  - Analog data acquisition and wireless transmission by Bluetooth
  - Automotive Glossary of Electrical, Electronic, & Communication terms

- Completed the following Hardware and Software Client Projects
  - Five Wireless Projects – 1. Anti-jam GPS Receiver 2. mm-wave RADAR 3. DSP-based Wireless base band for WiCOMM-T, 4. WiMAX System Controller (IEEE802.16 - MAC/Network Management & MIB), 5. Zigbee (Wireless Sensor Networks) Software development
  - Five Medical Systems Projects – 1. Modernisation of Auto-Perimeter by touchscreen-based Inputs and Outputs 2. Hardware and Software for Toric Marker 3. Hardware development for Needle Positioner – Biopsy device 4. Hardware and Software development for a compact Tonometer 5. Hardware and Software for Auto-Refractometer / Keratometer
  - One Software Project – 1. TI Sitara's (AM5728) processor-based software (Embedded Linux, TI RTOS)
  - Three Auxiliary Projects – 1. Ambe CODEC 4020 2. Neon Programming 3. IMU Sensor ADIS16497
  - Two Automotive Projects – 1. Java based Electronic Spare Parts Catalogue Software 2. Java based Eicher Service Information
  - Several Matlab/Student Projects – 1. Two-sided impacts (Fortran77 to Matlab conversion), 2. Sophisticated Matlab String Utility Functions, 3. Inventory Control (cost/profit/holding time/shortage optimization), 4. Delta-Sigma Modulator, 5. Unmanned Rail-Road Crossing (Gate Module and Train Module), 6. Remote Temperature Monitoring and Power Automation for PBX Server Room, 7. Unit Commitment problem solution using shuffled leap frog algorithm, 8. Image Feature Extraction, 9. Abstracting Keywords from Hypertext Documents using Stemming Algorithm
  - Two Application Development Projects – 1. Java based MP3 Mixer, 2. S60/Symbian Mobile Application for Smart home

- Work experience
  - Nokia Mobile Phones, San Diego. USA.
    - OSE based CDMA Engine Platform for 3G Cellular / PCS Phone
  - Dialogic Corporation, an Intel Company. New Jersey. USA.
    - Computer Telephony and Fax Products (Voice Brick, Chameleon (D82U))
  - Philips Consumer Communication. New Jersey. USA.
    - IS-95 / J-STD-008 CDMA Standard based Cellular/PCS Phone
  - Hughes Software Systems. India.
    - Fixed-point implementation of MM-CELP CODEC
    - Real Time User Terminal for ICO Land Mobile Satellite Communication System
  - Industrial Consultancy and Sponsored Research, IIT Madras. India.

- DigiCom Experimenter
- Programmable Power Line Carrier Communication
- CorDECT Wireless in Local Loop System

Instructor's Educational Background

- M.S. (Signal Processing)
  - Indian Institute of Technology, Madras. India.
- B.E. (Electronics and Communication)
  - Madurai Kamaraj University, Madurai. India.
- Specialized Certificate in CDMA Engineering
  - University of California, San Diego. CA. USA.
- Several other Signal Processing and Communication Courses from
  - University of California, San Diego. CA. USA.
  - Northeastern University, Boston. MA. USA.