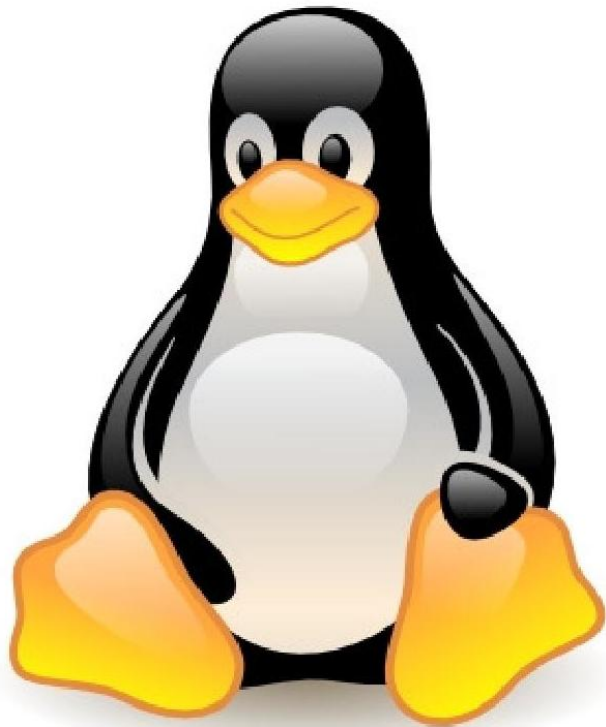


# Advanced Embedded Linux & Device Driver Development



## Training Highlights:

- Learn through Practical's
- Work on Latest ARM Core like Cortex A5/A8/A9
- Real World Examples and Projects
- Assured Post Training Support
- Unlimited Access to the Hardware Boards for Practical's
- Training Tutorials & data available online

# Module-1 : Linux Basics & Internals

**Duration: 4 Days**

**Timings: 9:30 AM to 5:30 PM**

S.no	Topic	Activities
1	<b>Linux Introduction &amp; Installation</b>	<ul style="list-style-type: none"><li>➤ What is Linux, How it has been evolved, GNU License &amp; Kernel</li><li>➤ How Linux was designed,</li><li>➤ Sub systems of Linux [ Scheduler, Process, Memory Management, File System, Device Management]</li><li>➤ Ways to Install Linux [1. Dual Boot, 2. Within Windows, 3. Using Virtual Machine ]</li><li>➤ How to update Linux and install required package</li></ul>
2	<b>Linux Shell Commands</b>	<ul style="list-style-type: none"><li>➤ Basic Commands</li><li>➤ Dir &amp; File Command</li></ul>
3	<b>Linux Shell Commands</b>	<ul style="list-style-type: none"><li>➤ System Commands</li><li>➤ Misc Command</li></ul>
4	<b>Shell Scripting</b>	<ul style="list-style-type: none"><li>➤ Writing Basic Linux Shell scripting</li><li>➤ Variables &amp; Operators in Shell scripting</li><li>➤ Command Line Arguments</li><li>➤ Logical Structures in Shell Scripting</li></ul>
5	<b>C Programming in Linux</b>	<ul style="list-style-type: none"><li>➤ Writing C program on Linux</li><li>➤ Compiling and executing Linux</li><li>➤ Linux Executable format info &amp; tools</li><li>➤ Debugging C application on Linux using GDB</li></ul>
6	<b>Makefiles</b>	<ul style="list-style-type: none"><li>➤ Understanding Makefiles</li><li>➤ Writing Makefiles</li><li>➤ Compiling Multiple source directory's using Makefile</li><li>➤ Advanced methods used in writing Makefiles</li></ul>

# Module-1 : Linux Basics & Internals

S.no	Topic	Activities
7	<b>Process Management</b>	<ul style="list-style-type: none"><li>➤ Understanding Linux Process</li><li>➤ How to create child process using [ system, exec, fork &amp; clone ]</li><li>➤ Managing Linux process</li></ul>
8	<b>File Operation in Linux</b>	<ul style="list-style-type: none"><li>➤ How to write application to access files in Linux</li><li>➤ System Calls used in Linux to control special files like device nodes</li><li>➤ How to write a serial port access program in Linux</li></ul>
9	<b>Linux Multi-Threading Programming</b>	<ul style="list-style-type: none"><li>➤ Basics of Multithreading in Linux</li><li>➤ How to create multi-threading applications in Linux</li><li>➤ Managing &amp; communication between Multiple threads</li></ul>
10	<b>Inter Process Communication</b>	<ul style="list-style-type: none"><li>➤ Data sharing between multiple processes using IPC Mech.</li><li>➤ Writing apps using PIPEs, FIFOs, Msg Queues, Shared Memory</li></ul>
11	<b>Network Programming in Linux</b>	<ul style="list-style-type: none"><li>➤ How to develop client server based network application in Linux</li><li>➤ When and how to use TCP and UDP Protocols</li></ul>
12	<b>Programming &amp; Debugging Tools</b>	<ul style="list-style-type: none"><li>➤ strace : Tracing System calls</li><li>➤ ltrace : Tracing Library calls</li><li>➤ Tools used to detect memory access error and Memory</li><li>➤ leakage in Linux : mtrace, valgrind</li><li>➤ Using gdb and ddd utilities</li><li>➤ Core Dump Analysis etc.</li></ul>

# Module-2 : Embedded Linux Porting

**Duration: 2 Days**

**Timings: 9:30 AM to 5:30 PM**

S.no	Topic	Activities
1	<b>Introduction, Setup &amp; Hardware</b>	<ul style="list-style-type: none"><li>➤ Introduction to Embedded Linux</li><li>➤ ARM Processor Basics &amp; Families</li><li>➤ ARM Board Details and Schematic Overview</li><li>➤ Boot Process</li><li>➤ Host PC Setup for eLinux Development</li></ul>
2	<b>Toolchain &amp; Hardware Practical's</b>	<ul style="list-style-type: none"><li>➤ Board Boot Options</li><li>➤ Flashing Bootloader &amp; Linux Kernel on Board</li><li>➤ Setting up TFT and Running Application on Board</li><li>➤ Toolchain &amp; its components</li><li>➤ How to build toolchain</li></ul>
3	<b>Bootloader U-Boot</b>	<ul style="list-style-type: none"><li>➤ Introduction to Bootloader</li><li>➤ Primary Bootloader ( TI X-Loader )</li><li>➤ Bootloader Commands and their usage</li></ul>
4	<b>U-Boot Porting</b>	<ul style="list-style-type: none"><li>➤ Bootloader Source Code Structure</li><li>➤ Compiling Bootloader</li><li>➤ How to port Bootloader on ARM Based Hardware</li><li>➤ Patching Bootloader</li></ul>
5	<b>Customizing Bootloader</b>	<ul style="list-style-type: none"><li>➤ Modifying Bootloader for new feature</li><li>➤ Modifying Bootloader to support new device</li><li>➤ Command Line Arguments &amp; ATAG</li><li>➤ Booting with SD Card</li><li>➤ Setting up NFS Server</li><li>➤ Booting with NFS Server</li><li>➤ Linux Kernel Compilation</li></ul>

# Module-2 : Embedded Linux Porting

S.no	Topic	Activities
6	<b>Linux Kernel</b>	<ul style="list-style-type: none"><li>➤ Introduction to Linux Kernel Arch</li><li>➤ Kernel Dir Structure</li><li>➤ Kernel Layers H/W dependent and independent ( BSP )</li><li>➤ Kernel Build System ( KConfig )</li></ul>
7	<b>Kernel Porting &amp; Compilation</b>	<ul style="list-style-type: none"><li>➤ How to configure and compile for ARM Hardware</li><li>➤ Type of kernel images ( vmlinux, zImage, uImage )</li><li>➤ Kernel initialization process</li><li>➤ How to port Kernel on New ARM Hardware</li></ul>
8	<b>Kernel Modification</b>	<ul style="list-style-type: none"><li>➤ How to modify the Kernel code</li><li>➤ How to integrate new driver / module in kernel image</li><li>➤ Building static and dynamic kernel modules</li></ul>
9	<b>Root File System</b>	<ul style="list-style-type: none"><li>➤ Components of Roofs</li><li>➤ Types of Roofs</li><li>➤ Different types of Flash Device ( NOR / NAND )</li><li>➤ Building Roofs from scratch and using Build System ( Buildroot )</li></ul>
10	<b>Embedded Application Development</b>	<ul style="list-style-type: none"><li>➤ How to develop embedded applications</li><li>➤ Debugging application on target using GDB</li><li>➤ Running sample Web-Server Application</li><li>➤ Using Eclipse for embedded application development</li></ul>

# Module-3 : Device Driver Development

**Duration: 2Days**

**Timings: 9:30 AM to 5:30 PM**

S.no	Topic	Activities
1	<b>Introduction and Arch of Linux Device Drivers</b>	<ul style="list-style-type: none"><li>➤ Introduction to Kernel Space and User Space</li><li>➤ Memory Management in Kernel</li><li>➤ How to develop Kernel Device Driver</li><li>➤ Layers of LDD</li><li>➤ Processor Memory Layout</li><li>➤ Device Register Access from Code</li></ul>
2	<b>Kernel Module Programming</b>	<ul style="list-style-type: none"><li>➤ Kernel Module Programming</li><li>➤ Module Parameters</li><li>➤ Exporting Symbols between modules</li></ul>
3	<b>Character Device Drivers</b>	<ul style="list-style-type: none"><li>➤ Linux Kernel Device Driver Framework</li><li>➤ Virtual File System as bridge between Driver and Application</li><li>➤ Implementing basic character driver</li></ul>
4	<b>Character Device Drivers</b>	<ul style="list-style-type: none"><li>➤ Writing Makefile to compile Device driver</li><li>➤ Compiling and running on X86</li><li>➤ Cross Compiling and running on ARM Hardware</li></ul>
5	<b>Advance options in Character Device Drivers</b>	<ul style="list-style-type: none"><li>➤ Implementing advance api like ioctl in character device driver</li><li>➤ Standards to follow while implementing ioctl</li><li>➤ Writing and testing LED driver with IOCTL on ARM Hardware</li></ul>
6	<b>Interrupts in Device Driver</b>	<ul style="list-style-type: none"><li>➤ Interrupts in ARM Processor</li><li>➤ Interrupts Mechanism in Linux Kernel</li><li>➤ How to implement Interrupts in device driver</li></ul>
7	<b>Interrupt Handling &amp; Bottom Half</b>	<ul style="list-style-type: none"><li>➤ Writing and testing Interrupt for Button press on ARM Target</li><li>➤ Writing and testing multiple Interrupts in single driver</li><li>➤ How to implement Shared Interrupts</li><li>➤ How to handle lengthy ISR using Bottom Half ( Soft IRQ, Tasklet &amp; Workqueues )</li></ul>

# Module-3 : Device Driver Development

S.no	Topic	Activities
8	<b>Special File Systems ProcFS &amp; SysFS</b>	<ul style="list-style-type: none"><li>➤ Ram based files systems in Linux</li><li>➤ Using procs for special purpose and accessing kernel data structure</li><li>➤ How to implement procs</li><li>➤ Sysfs implementation in device drivers for easy application access.</li></ul>
9	<b>Introduction to Block Device and Network Device Drivers</b>	<ul style="list-style-type: none"><li>➤ Introduction to block and network device drivers</li><li>➤ Case study of Network Device Drivers</li></ul>
10	<b>Advance Device Drivers and debugging</b>	<ul style="list-style-type: none"><li>➤ MTD Subsystem for Flash Memory Devices</li><li>➤ Nand and Nor Device Drivers</li><li>➤ USB Subsystem Introduction</li><li>➤ How usb gadget drivers are used in Embedded Applications</li><li>➤ Debugging Techniques like debugs / target debugging</li></ul>

## Optional:

- If you are having any technical queries related to syllabus covered (LAB & Theory), we provide an additional weekend session for the same.

# Embedded Linux Training With Free Hardware Kit

Embedded Linux  
Porting

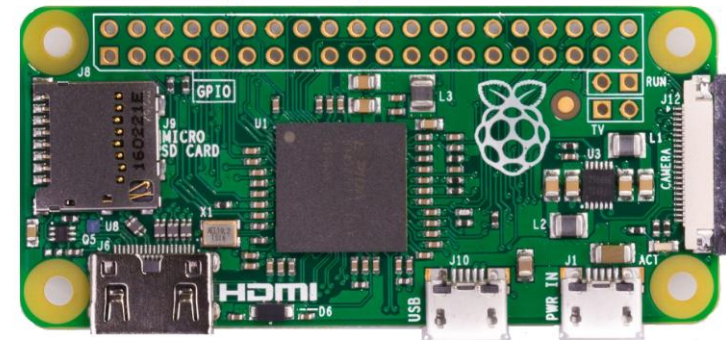
Linux Basics &  
Internals

Device Driver  
Development

+

Free Hardware

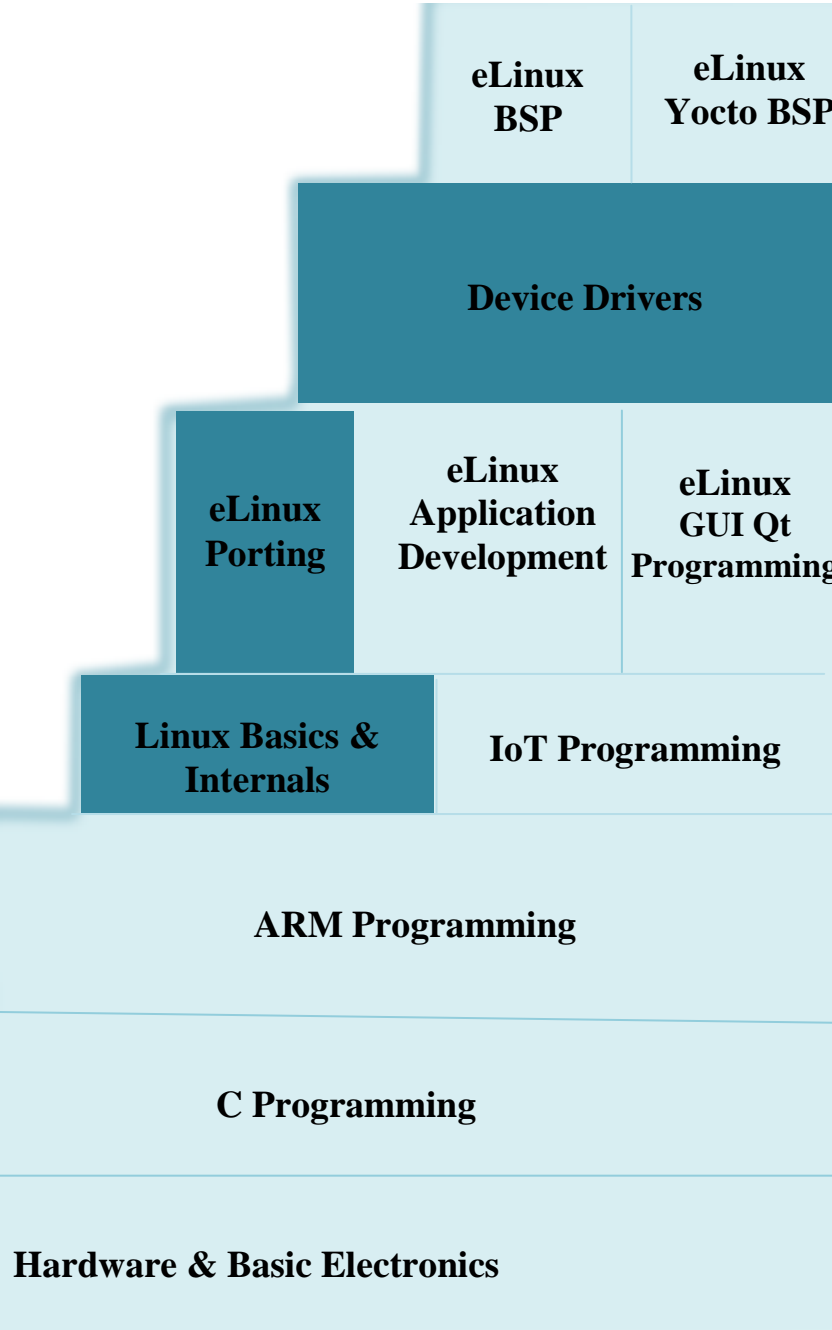
Raspberry PI Zero





**Embedded Engineering Steps**

**Locations / Fees / Duration**



**Embedded Engineering**

**Locations**

- Bangalore

**Duration:**

- 8 Days

**Fees:**

- 1800.00 INR
- [Register Now](#)

**After development workshop:**

- When you return to work, you are entitled to schedule a technical discussion with the course instructor for help and guidance as you apply your new skills to your projects.

**Address:**

- AESLAB | #9/1 1st Floor, 3rd Main, 8th Block, Opp. Police Station, Kormangala, Bengaluru, Karnataka 560095
- Email: [info@easyarm.com](mailto:info@easyarm.com), Phone: +91-80-41307589 || +91-9972039671, Web: [www.aeslab.com](http://www.aeslab.com)