

Embedded Linux Training (60Hrs)



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 [<http://learn.easyarm.com>]

www.easyarm.com || info@easyarm.com || +91-80-41307589 || +91-9972039671

Embedded Linux Learning 60Hrs [Rs. 12000.00]

Linux Internals (Module-1)

Introduction to Linux

- GNU Project / GPL Licensing
- Evolution of Linux & Development Model
- Device Identities in Linux - Partitioning Schema

Introduction to Kernel

- History of Linux - Types of Kernel - The Linux kernel
- Kernel Architecture

Shell Commands & Shell Scripting

- Basic Shell commands - Bash Shell Essentials
- Creating Makefiles

Creating Libraries

- Creating Static Library - Creating Shared Library

The Boot Process

- BIOS Level - Boot Loader - Setup, startup_32 functions
- The start_kernel() function

The File System

- Virtual File system & its role
- Files associated with a process - System Calls

Process management

- Process Defined
- Process Descriptor Structures in the kernel
- Process States - Process Scheduling
- Process Creation - System calls related to process management

Memory Management

- Defining and Creating secondary memory areas
- Memory allocation & deallocation system calls malloc, calloc, alloca, free
- Demand Paging defined
- Process Organization in Memory
- Address Translation and page fault handling
- Virtual Memory Management

Multi Thread Programming

- Creating multiple threads
- Parent synchronization with other Threads

Inter Process Communication

- Pipes, Fifo's, signals - System-V IPC's
- Message queues - Shared memory - Semaphores

Sockets

- An Overview
- System calls related to TCP and UDP sockets

Network Programming

- TCP Server Client Programming
- UDP Server Client Programming
- Netlink socket interface

Programming & Debugging Tools

- strace : Tracing System calls
- ltrace : Tracing Library calls
- Tools used to detect memory access error and Memory leakage in Linux : mtrace
- Using gdb and ddd utilities
- Core Dump Analysis etc

Building Embedded Linux System (Module-2)

Lecture: Introduction to Embedded Linux and Blueboard-AT91RM9200

Lecture:

- Toolchain Components
- Building Toolchain
- Build Systems for compiling toolchain

Lab: Toolchain compilation and usage.

Lecture:

- Bootloader Architecture
- U-Boot Bootloader Porting on New Hardware.
- U-Boot Commands

Lab: Bootloader compilation and downloading on Target board.

Lab:

- Bootloader commands and usage,
- Bootloader code customization, adding new Ethernet drivers to U-Boot.

Lab: Downloading pre-compiled Linux kernel images on Target board.

- > Using SD-Card for rootfs.
- > Configuring NFS and using rootfs over NFS.
- > Configuring TFTP and downloading kernel image over TFTP.

Building Embedded Linux System-II

Lecture: Linux Kernel Architecture.

Lecture: Linux Source code browsing & code changes for a new Target [Porting]

Lab: Configuring and compiling Linux Kernel.

Lecture: Root file system.

Lab: Building Root file system with Busybox and booting the Linux Kernel

Lab: Cross compilation of libraries for target.

Lab: Application development and Cross compilation.

Linux Device Drivers - (Module-3)

An introduction to device drivers

- Role of the Device Drivers
- Splitting the kernel
- Classes of devices and modules

- Kernel Architecture or Model
- kernel modules

Module Basics

- Introduction to Modules & Device Drivers

- Modules Defined
- Types of Modules in the kernel
- Writing Your first kernel module
- Module Related Commands
- Kernel Module vs Applications
- User space vs Kernel space
- Statically linked vs Dynamically linked drivers/modules
- Exporting symbols from modules
- Concurrency in the kernel
- Module Parameters
- Version dependency
- Kernel Module Programming
- Lab exercises

The proc file system

- Creating proc file system entries
- Making read & write operations on proc entries
- Lab exercises

Character Device Drivers

- Registering a character device driver
- File operations
- The file structure
- devfs / lseek / ioctl
- Blocking, non blocking and asynchronous operations
- Programming with ioctl(), mmap()
- Lab exercises

Linux Device Drivers -II

Hardware and Interrupt Handling

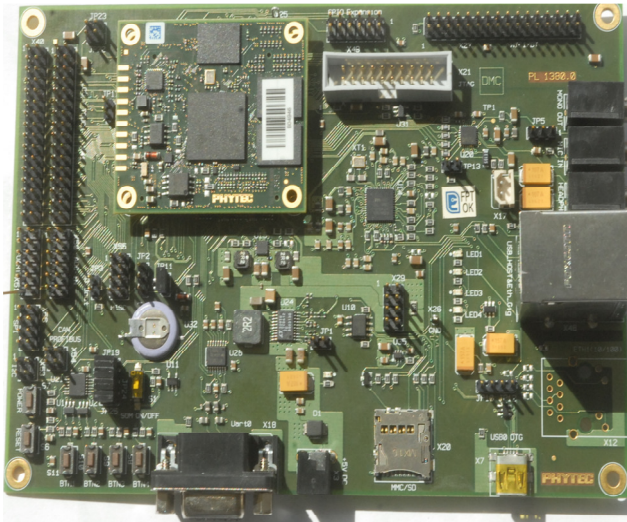
- Using IO Ports
- Installing and implementing an interrupt handler
- Tasklets and Bottom halves
- Task queues, Work queues - Lab exercises

Block Device Drivers

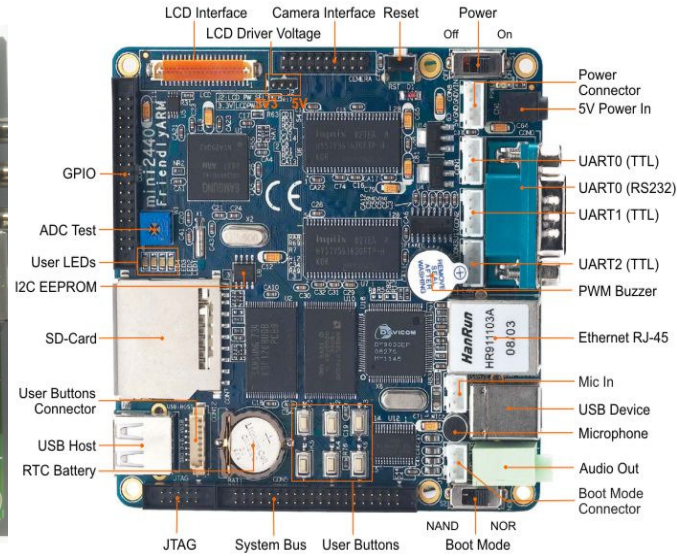
- Block drivers structures - Flash memory Drivers
- Lab exercises

Network Drivers

- The net_device structure in detail
- Making changes in Ethernet drivers in kernel Source.
- Lab exercises



PHYTEC-OpenBoard [Rs. 4800.00]



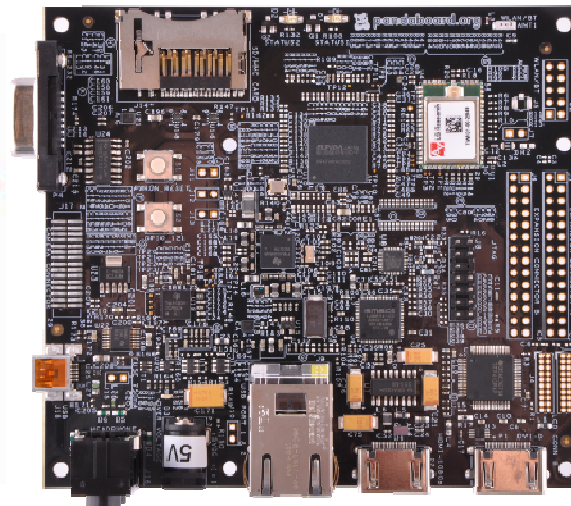
Mini2440 [Rs. 5500.00]



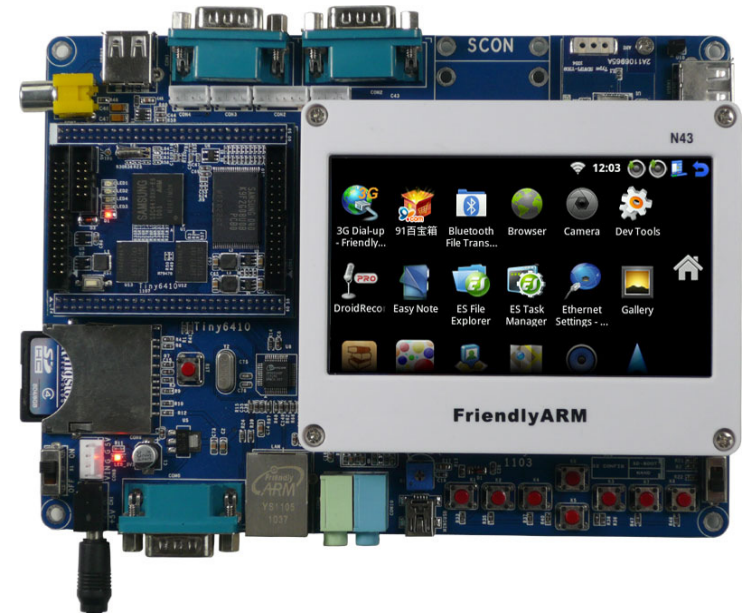
HawkBoard [7000.00]



BeagleBoard [Rs. 9800.00]



PandaBoard [Rs. 14500.00]



Tiny-6410 [Rs. 8500.00]