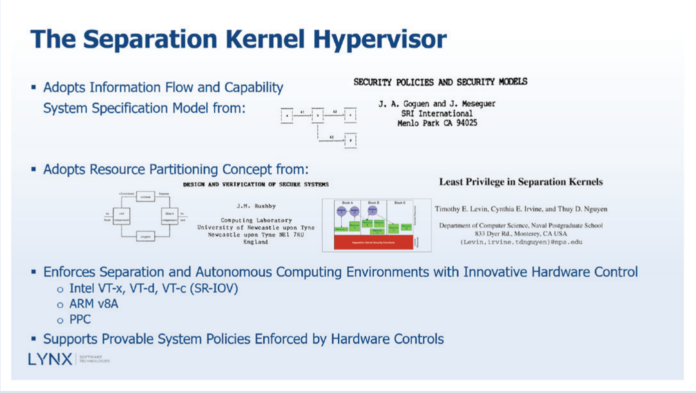
LYNX MOSA.ic Technical Training

Background

Project team productivity is critical to meeting program timelines and budgets. Lynx offers three, four, and five-day training courses to ramp your engineering team, instill best practices, and maximize their potential. Class sizes are limited to five to eight students and provide each student with access to a hands-on environment to complete the class exercises. The days are broken down into half-day 4 hour sessions. The last half-day is allocated to custom content development.

The objective of the training class is to enable customers to understand how best to harness all the features and capabilities of this software framework.

In terms of class approach, there is structured content delivery (via Webex) and hands-on labs (via NoMachine). The student needs only a computer and Internet connection capable of connecting to Lynx's remote training lab, although an additional monitor is recommended for multiple screens. Schedules do vary based on workloads but classes are typically scheduled 3-4 weeks after receiving the booking.

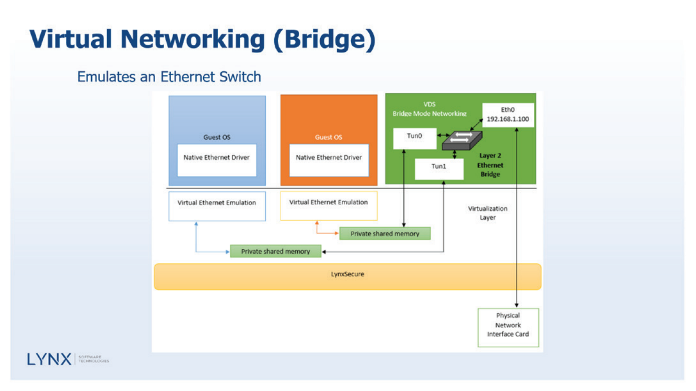


Course Outline

LYNX MOSA.ic for Avionics (MfA) is a safety certifiable RTOS platform with integrated hypervisor built to enable the Modular Open Systems Approach (MOSA) as defined by the US DoD.

MfA consists of:

- LynxSecure; Separation Kernel Hypervisor
- LynxOS-178; Safety certified RTOS
- Buildroot; Embedded Linux
- LSA; Lynx Simple Application, bare-metal operating environment



Topic 1: LYNX MOSA.ic Introduction

- Introduction to LYNX MOSA.ic for Avionics

- LynxSecure Overview
  - Design Overview and Theory; This provides an introduction to the research behind LynxSecure and high-level concepts that are important to understand when building a safe and secure system.
  - Feature Overview; A high-level introduction to the features of LynxSecure and what capabilities are required of the hardware to build a safe and secure system.

- LynxOS-178 Overview
  - DO-178 Background
  - Architectural and Feature Overview

Topic 2: LynxSecure deep dive

- Development Environment; Describes the components provided in the LynxSecure software development kit (SDK), where they are installed, overview of the LynxSecure development process, and an introduction to using the license-management software. Students will use this knowledge to navigate within their LynxSecure project trees.

- Autoconfig Basics; Introduction to using the LynxSecure "autoconfig" tool and its syntax, creating target system configurations, and the contents of the system runtime package (SRP) that will be booted on the target system. Studentswill use this knowledge when working with the LynxSecure tools to create models for their target systems describing how resources will be allocated and what permissions are assigned to which rooms.

- Virtual Device Server (Intel architecture only).

- Overview of the Virtual Device Server (VDS) and what services it can provide out of the box.

- Virtual HDD, CD, KVM; Describes how to partition a virtual hard disk so VDS can create virtual disks, how virtual CDROM drives can be attached to a room and how to use VDS' virtual keyboard, video, and mouse (KVM) capabilities to provide KVM services to multiple rooms on a single target system.

- Virtual Networking; A detailed description as to how to configure LynxSecure's virtual networking for the following use cases;
  - Configuring point-to-point virtual Ethernet connections.
  - Configuring VDS to NAT virtual Ethernet devices to a physical Ethernet NIC.
  - Configuring VDS to bridge virtual Ethernet devices across a physical Ethernet NIC.

- Buildroot Linux Basics; An Introduction to Buildroot Linux and managing targets and using templates. This section also shows how to make changes to user space, kernel space, and busybox in Buildroot. It also describes how to add custom packages to Buildroot.

Topic 3: LynxSecure Advanced Features

- Scheduling; How to configure LynxSecure's scheduling policies; how to use flexible scheduling, core time dona- tion; concepts that can be used to create custom core scheduling solutions using time donation.

- Fully / Para Virtualization; This section compares and contrasts fully virtualized rooms vs. para-virtualized rooms so an appropriate room type can be made for each room.
  - HCV, BCV, SRP + Boot Process; Provides an explanation of the human-readable configuration vector (HCV) and how to fine tune it so developers can get the exact configuration they desire. This section also discusses the binary configuration vector (BCV) that is produced when the HCV model is processed. There is a break down of the components contained within the SRP, including the BCV and relates the SRP's components to its boot process after control is handed to the SRP from the bootloader.

- Multicore + SMP; A quick introduction to LynxSecure's ability to run multiple guests on multiple cores – both single core rooms and multi-core rooms.

- Hypercalls; Provides an introduction to hypercalls, discusses the categories of hypercalls and what they do, and provides details as to how to add permissions for hypercalls into the HCV model.

- Time; Discusses hypercalls related to system clock management. Shows how to add clock flows to allow a room to read and/or write the system clock and dis- cusses absolute clock time and monotonic clock time.

- State Engine; A discussion of the states that a room can be in and how to transition from one state to another.

This section also covers how hypercall permissions are checked by the policy decision engine (PDE). Finally, this section covers the system state manager (SSM), the target system's states and what state transitions can be made.

- Interrupts; An introduction to the concept of interrupt controllers and different types of interrupts.

- Message Channels; An introduction to message channels as an inter-room communications device.

- Shared Memory + Synthetic Interrupts; Showing how to use shared memory and synthetic interrupts as inter-room communications mechanisms.

- Audit; An in-depth discussion of LynxSecure's auditing facilities.

- Built-In Test; Details of LynxSecure's built-in test (BIT) capabilities and what it tests.

- Lynx Simple Applications; Introduction to Lynx Simple Applications (LSAs), which are bare-metal applications.

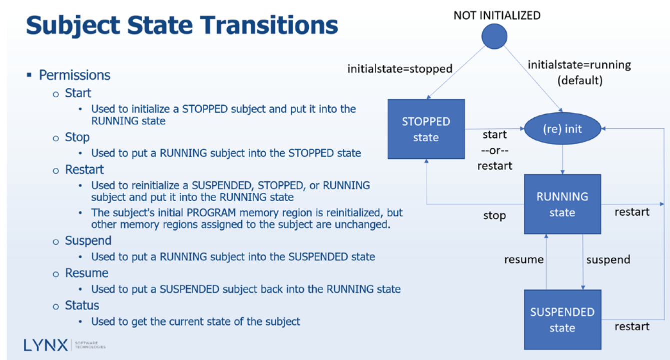
Topic 4: LynxOS-178

The objective of this section is to teach the attendees how to write real-time applications; how to harness POSIX and ARINC applications with LynxOS-178; understand from an application point of view how LynxOS-178 works from the perspective of processes and threads, address spaces, inter-process and inter-thread communication, as well as inter-partition/intra-partition perspectives. It also reviews how to harness the development tools provided with LynxOS-178.

- Introduction
  - Recommended Reading
  - Supported Standards

- Architecture
  - CPU Support Package (CSP)
  - Board Support Package (BSP)
  - Device Resource Manager (DRM)
  - System Services
  - Startup Conditions and Schedule
  - Master Process
  - ARINC 653 Partitioning

- Cross Development Environment
  - Cross Development Kit (CDK) Contents
  - Production Mode versus Development Mode
  - License Management
  - Luminosity (Eclipse based IDE)
  - Training class enables attendees to create a LynxOS-178 kernel, register a target development platform, create a "Hello World" application and debug the application
  - SpyKer (System Trace and Optimization Tool)
  - Show how to capture and analyze a system's performance characteristics
  - Simple Kernel Debugger (SKDB)
  - Outlining how this is a machine-level symbolic debugger is used to support debugging of LynxOS-178 kernel internals, including device drivers by setting breakpoints, examining memory and registers, changing memory contents and displaying kernel data structures
    - Kernel Configuration
    - BSP Components
    - Virtual Machine Configuration Table (VCT)
    - File System



IEEE 754 Floating Point Support

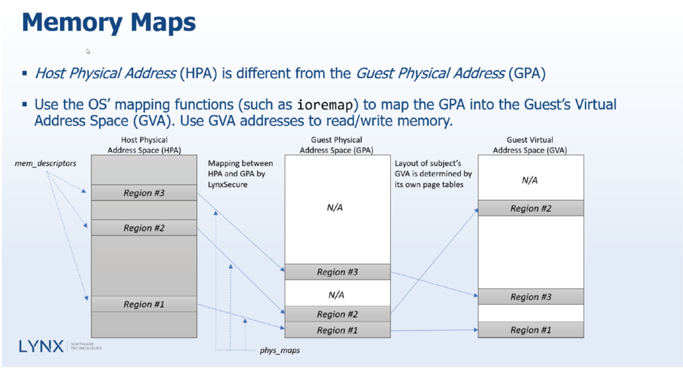
- Processes
  - Processes Definition
  - Virtual Address Space
  - The ps command
  - States
  - Process management (startup, waiting, exiting, user IDs)

- Threads
  - Thread model
  - Thread safe functions and libraries
  - Thread creation and cancelation
  - Thread vs. Process attributes

- Scheduler
  - Terminology
  - Thread States
  - Scheduling Model and Ready Queues
  - Priority and Policy

- Pipes
  - Using named and unnamed pipes
  - Non-Blocking Mode with Pipes
  - Atomicity of Pipe Writes
  - Design Considerations

- Message Queues
  - Definition
  - Naming Convention and Attributes
  - Using Message Queues
  - Message Priorities



- Standard Signals
  - This section reviews the use of these software interrupts that a thread uses to send a signal to itself or other threads. This section will review how these can be used for functionality such as Process control, thread synchronization, basic data exchange and exception handling

- Real Time Signals
  - A review of how these signals are queued, how they pass data and how they are prioritized

- Date/Time
  - A review as to how LynxOS-178 Handles inputs from onboard real-time clocks (RTC) and System Clocks

- Timers
  - Creating and setting a Timer
  - Retrieving Time Left on a Timer
  - Handling timer expiration and overruns
  - Absolute vs. Relative Timer

- Shared Memory
  - Creating, Opening, and Mapping
  - Getting and Setting Size
  - Unmapping, Closing, and Removing

- Semaphores
  - Named and Unnamed Semaphores
  - Creating, Opening, Using, Closing, and Removing
  - Retrieving a Semaphore's Value

- Mutexes, Condition Variables and Barriers

- Filesystem I/O
  - LynxOS-178 File I/O
  - Synchronized I/O and Flags

- ARINC 653 APEX
  - Configuration
  - Return Code Datatype
  - Sampling Ports
  - Queuing Ports

- Lynx Certifiable Stack (LCS)
  - Features
  - Configuration
  - Global Configuration Considerations

Topic 5: Custom Development

1.800.255.5969

Lynx Software Technologies, Inc.
855 Embedded Way
San Jose, CA 95138-1018
+1 (800) 255-5969
+1 (408) 979-3900
+1 (408) 9793-920 fax
inside@lynx.com
www.lynx.com

Lynx Software Technologies UK
400 Thames Valley Park Drive
Thames Valley Park
Reading, RG6 1PT
United Kingdom
+44 (0) 118 965 3827
+44 (0) 118 965 3840 fax