KeyStone 2 Multicore Workshop:

**Preliminary session – 5.5 hours**

The purpose of this session is to introduce to students that are not familiar with CCS and SYSBIOS the two basic building blocks of TI DSP systems

* Introduction to Code Composer Studio (CCS) Overview (60 minutes) examines the Eclipse-based CCS IDE from TI, with an emphasis on the new elements incorporated into CCS Version 5.x, including code generation tools, compiler, linker, map file.
* LAB 0.1: CCS Basics Part One (60 minutes)
  + Import Example Project: SRIO Direct IO Loopback
  + Create Target Configuration
  + Connect/Load/Build/Run
  + Create a new CCS project: Optimization (Tasks1-3)
  + Connect/Load/Build/Run
* Introduction to KeyStone 2 EVM (30 minutes)
* KeyStone 2 Out-of-the-box demo- Lab 0.2 (60 minutes)
* Introduction to SYS/BIOS (60 minutes)
* Multicore EDMA Usage (60 minutes) takes a look at different DMA methods used on the C66x including EDMA3, QDMA, and IDMA. It provides information on programming, linking, and chaining EDMA3, examples of the transfer and sorting functions.

**Session 1 Basics – 6.5 hours**

This is the first workshop session. We will cover the basic hardware and software architectures

* KeyStone (I and II) Device Overview (75 minutes) introduces the C66x SoC functional architecture including the CorePac, Memory Subsystem, internal transport mechanisms, external interfaces, accelerators and coprocessors, and other miscellaneous features.
* C66 CorePac (60 minutes) provides a more detailed description of the C66x CorePacs including the functional units, internal and external interfaces, interrupt controller, etc.
* ARM CorePac Overview (60 minutes) provides a more detailed description of the four ARM Cortex A15 CorePacs including the functional units, Neon and VFP, cache coherency, etc.
* KeyStone (I and II) Software Overview (60 minutes) introduces the suite of tools provided by TI and third-party partners to enable application development on C66x SoC devices, including Code Composer Studio (CCS), the Multicore Software Development Kit (MCSDK), third-party plug-ins, and the C66x Lite Evaluation Module (EVM). In addition, ARM Software Overview presents the software structures (Linux, SysLink, LLD) that can be called from a Linux-run ARM.
* **KeyStone 1+2 Peripherals usage (75 minutes). This presentation covers the software support for the various IP and peripherals. It covers resource management, the CSL layer structure and examples of API, LLD structure, examples and API from the C66 and the ARM cores, understanding higher layer of LLD such as the NetCp.**
* **Lab 1 (60 minutes)– Peripheral LLD Lab or demo + ARM**

**Session 2 Communication and Peripherals – 6.5 hours**

In this session we cover the hardware and software mechanism to move data within and outside the device

* Memory & Cache Considerations (60 minutes)
* Introduction to Interprocessor Communication (60 minutes) provides an overview of basic IPC concepts, compares and contrasts different services  
  provided within the IPC framework, analyzes support utilities, describes configuration, and provides a few IPC usage examples.
* **LAB: IPC Shared Memory (60 minutes) + ARM**
* Multicore Navigator Usage (90 minutes) provides an overview of the hardware mechanism that facilitates data movement and multicore cooperation in KeyStone SoC devices. Topics include the Navigator subsystem architecture, use cases and example code, configuration, and low level drivers.
* **LAB: IPC Multicore Navigator Transport (30 minutes) + ARM**
* PCIe Overview with Lab/demonstration (90 minute)

Or

* HyperLink Overview with Lab/demo (90 minutes)

**Session 3 System Issues – 7.5 hours**

In this session we cover system issues including debug and boot-loader

* Multicore Design Considerations (75 minutes) introduces the concepts of parallel programming and processing and illustrates their implementation through video-encoding use case scenario- including VLFFT demonstration.
* C66 Single-core Optimization (75 minutes) including pipeline considerations, software pipeline and how to achieve it, and other useful optimization techniques
* LAB: Optimization (60 minutes)
  + Compiler Optimization
  + Enable Software Pipeline
  + Align Data
  + Cache Considerations/Analysis
* Bootloader (90 minutes) provides an introduction to the C66x + ARM bootloader including configuration, device startup, and runtime modes. Including a Lab/demo
* C66x Advance Debug/Trace (60 minutes) includes the following demos/labs:
  + Basic debug
  + Usage of STM library for real-time printf (based on SRIO demonstration)
  + Demonstrate usage of debug module instrumentation
* ARM Advance Debug/Trace (60 minutes) includes the following demos/labs:
  + Basic debug
  + Usage of ARM library for real-time debug
  + Demonstrate usage of debug module instrumentation
* Advanced topic (1 hour) such as:
  + NetCP Overview including PA demonstration

Or

* + openMP

Or

* + SRIO Usage Overview and Lab