KeyStone Multicore Software Development Ecosystem
Agenda

• Multicore Development Ecosystem
  – Code Composer Studio (CCS)
  – Multicore Software Development Kit (MCSDK)
  – Third Party Software
  – C66x Lite Evaluation Module (EVM)
TI Software Development Ecosystem
Multicore Performance, Single-core Simplicity

Eclipse IDE

- Code Composer Studio™
- Third Party Plug-Ins

Multicore Software Development Kit (MCSDK)

Host Computer

Target Board

Emulator
TI Software Development Ecosystem
Code Composer Studio + Eclipse IDE

Eclipse IDE
- Code Composer Studio™
- Third Party Plug-Ins

Multicore Software Development Kit (MCSDK)

Host Computer

Target Board

Emulator
Code Composer Studio (CCS)

- Code Composer Studio (CCS) is an Eclipse-based IDE that supports application development on multiple cores/devices:
  - GUI interface for SYS/BIOS, project-based system to build drivers and utilities for developer’s target platform.
  - Multiple perspectives (default and custom)
  - Advance debugging, monitoring, and profiling
  - Multiple configurations allow a single executable or multiple executables to be generated for the same project.

- Editor
- Integrated compiler tools
  - Support for OpenMP
- Simulator
- Debug/Emulation
- Remote Debug
- Instrumentation
- Visualization
TI Software Development Ecosystem
Multicore Software Development Kit

Eclipse IDE

Code Composer Studio™
- Editor
- Compiler Tools
- Simulator
- Debug/Emulation
- Remote Debug
- Instrumentation
- Visualization

Third Party Plug-Ins

Host Computer

Emulator

Target Board

Multicore Software Development Kit (MCSDK)
MCSDK Overview

- Standard set of APIs to configure and utilize peripherals, accelerators, and other hardware resources
- Compliant implementation of packet- and network-based protocols
- Utilities to boot, test, debug, and monitor execution
- Individual core-based real-time operating system
- Communication and facilitated cooperation between processes, cores, and devices, as well as between peripherals and cores
- Optimized, ready-to-use algorithm libraries, example code, and demonstration applications
MCSDK Advantages

• Provides the core foundational building blocks for customers to quickly start developing embedded applications on TI high-performance multicore DSPs:
  – Uses either the SYS/BIOS or Linux real-time operating system
  – Accelerates time-to-market by focusing on ease of use and performance
  – Provides multicore programming methodologies and utilities

• Simplifies porting of applications:
  – To a standard evaluation platform
  – From a standard evaluation platform to customer’s target platform
  – To next generation platform hardware

• Available as a free download on TI.com, bundled in one installer as source code along with pre-built libraries
BIOS-MCSDK Software Layers

Demonstration Applications
- HUA/OOB
- IO Bmarks
- Image Processing

Software Framework Components
- Inter-Processor Communication (IPC)
- Instrumentation (MCSA)

Communication Protocols
- TCP/IP Networking (NDK)

Algorithm Libraries
- DSPLIB
- IMGLIB
- MATHLIB

Platform/EVM Software
- Platform Library
- Transports - IPC - NDK
- Resource Manager
- Power On Self Test (POST)
- OS Abstraction Layer
- Bootloader

Low-Level Drivers (LLDs)
- EDMA3
- PA
- SRIO
- FFTC
- TSIP
- PCIe
- QMSS
- CPPI
- HyperLink
- …

Chip Support Library (CSL)

Hardware
## Development Requirements > MCSDK Solutions

<table>
<thead>
<tr>
<th>Development Requirement</th>
<th>MCSDK Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hide hardware details from the user to simplify process of porting to new hardware.</td>
<td><strong>Chip Support Library (CSL)</strong> is the only MCSDK layer that depends on the hardware. This layer is completely transparent to the user/application.</td>
</tr>
<tr>
<td>Standard API to talk to peripherals, accelerators, and other resources</td>
<td><strong>Low Level Drivers (LLD)</strong> provide standard API to initialize, configure, and utilize peripherals and other resources. LLD blocks include SRIO, PCIe, PA, CPPI, QMSS, FFTC, and many more.</td>
</tr>
<tr>
<td>Utilities to facilitate system operations</td>
<td><strong>Platform/EVM Software</strong> provides platform-level utilities such as bootloader, Power On Self Test (POST), resource manager, and platform utilities.</td>
</tr>
<tr>
<td>Efficient real-time individual core operating system</td>
<td><strong>SYS/BIOS</strong> provides an efficient, mature, real-time operating system with a low memory footprint.</td>
</tr>
<tr>
<td>Reliable interface to external networks</td>
<td><strong>Network Development Kit (NDK)</strong> provides a standardized interface for common packet- and network-based communication protocols (e.g., IPV4 and IPV6-compliant TCP/IP).</td>
</tr>
<tr>
<td>Coordination of tasks/process across multiple cores</td>
<td><strong>Inter-Processor Communication (IPC)</strong> provides several-high level utilities and libraries to communicate between cores and enable multiple cores to work together.</td>
</tr>
<tr>
<td>Facilitate application development</td>
<td><strong>Optimized algorithm libraries</strong> with standard APIs.</td>
</tr>
<tr>
<td>A starting point for multicore application development</td>
<td><strong>Demonstration applications</strong> (e.g., Image Processing) show how to build and run a complete multicore application.</td>
</tr>
</tbody>
</table>
Communication via LLD and CSL Layers

- Turbo Decoder (TCPD)
- Turbo Encoder (TCPE)
- LLD Layer
- CSL Function Layer
- CSL Registers Layer
MCSDK Algorithm Libraries

Algorithm libraries contain C66x C-callable, hand-coded, assembly-optimized functions for specific usage:

- **DSPLIB** provides signal-processing math and vector functions:
  - Adaptive filtering
  - Correlation
  - FFT (e.g. FFT functions for ‘npoint’ FFTs)
  - Filtering and Convolution (e.g., FIR, IIR filter functions, etc.)
  - Matrix (e.g., single and double precision matrix multiplication, etc.)

- **IMGLIB** provides image/video processing functions:
  - Compression & Decompression (e.g., forward and inverse DCT, motion estimation, quantization, etc.)
  - Image Analysis (e.g., edge detection, histogram, thresholding, etc.)
  - Image Filtering and Conversion (e.g., color space conversion, convolution, correlation, error diffusion, etc.)

- **MathLIB** provides floating-point math functions:
  - Single-precision (e.g., cosine/sine/tangent of a floating point number, etc.)
  - Double precision (e.g., similar functions as above with argument type and return values to be of type double)

Inter-Processor Communication (IPC) Usage

IPC provides a standard interface between processes/threads in the same core, between two cores, and between two devices.
Simplified Development & Migration

TI Demo Application on TI Evaluation Platform

TI Demo Application

Customer Application

Customer App on Next Generation TI SOC Platform

No modifications required

May be used “as is” or customer can implement value-add modifications

Needs to be modified or replaced with customer version

Software may be different, but API remain the same (CSL, LLD, etc.)
MCSDK Top-level Directory Folders

- /mcsdk
- /pdk
- /edma3
- /bios
- /dsplib
- /imglib
- /mathlib
- /ipc
- /ndk
TI Software Development Ecosystem
Third Party Software Plug-ins

Eclipse IDE
- Code Composer Studio™
  - Editor
  - Compiler Tools
  - Simulator
  - Debug/Emulation
  - Remote Debug
  - Instrumentation
  - Visualization

Third Party Plug-Ins

Multicore Software Development Kit (MCSDK)

- Demonstration Applications
  - HUAOB
  - IO Benchmarks

- Communication Protocols
  - TCP/IP Networking (NDK)

- Algorithm Libraries
  - DSPUB
  - IMGUB
  - MATHLIB

- Platform/EVM Software
  - Platform Library
  - Transports: IPC - NDK
  - Power On Self Test (POST)
  - Bootloader

- Low-Level Drivers (LLDs)
  - EDMA3
  - PR
  - SRIO
  - FFTC
  - T3IP
  - PCIe
  - QMSS
  - CPPI
  - HyperLink

- Chip Support Library (CSSL)

Host Computer

Target Board

Emulator

Multicore Training
Third Party Software Plug-Ins

Eclipse allows developers to integrate third-party software tools that provide additional multicore programming, profiling and analysis capabilities:

- **Poly-Platform from PolyCore** [http://polycoresoftware.com](http://polycoresoftware.com) is a development framework consisting of tools and run-time software that provide a programming model for applications to scale from one to many cores.

- **Prism from CriticalBlue** [http://www.criticalblue.com](http://www.criticalblue.com) provides multicore analysis and exploration tools to evaluate parallelization strategies of existing software applications.

- **Optima from ENEA** [http://www.enea.com](http://www.enea.com) includes overview and management tools for multicore systems, profiling tools showing resource usage, and debug tools that track execution of application and operating system events.

- **Diamond from 3L** [http://www.3l.com](http://www.3l.com) is a tool-suite and model that provides a highly automated development flow from concept through to applications running in multiprocessor hardware.
TI Software Development Ecosystem
Multicore Performance, Single-core Simplicity

Eclipse IDE
- Code Composer Studio™
  - Editor
  - Compiler Tools
  - Simulator
  - Debug/Emulation
  - Remote Debug
  - Instrumentation
  - Visualization
- Third Party Plug-Ins
  - PolyCore
  - Critical Blue
  - ENEA Optima
  - 3L

Multicore Software Development Kit (MCSDK)

Demonstration Applications
- HUAOB
- IO Emarks
- Image Processing

Software Framework Components
- Inter-Processor Communication (IPC)
- Instrumentation (MCSA)

Communication Protocols
- TCP/IP Networking (NDK)

Algorithm Libraries
- DSPUB
- IMGUB
- MATHLIB

Platform/EVM Software
- Platform Library
- Transports: IPC - NDK

Low-Level Drivers (LLDs)
- Resource Manager
- Power On Self Test (POST)
- OS Abstraction Layer
- Bootloader
- Chip Support Library (CGL)

Hardware

Host Computer
Emulator
Target Board

Multicore Training
Linux/BIOS MCSDK C66x Lite EVM Details

DVD Contents

- Factory default recovery
  - EEPROM: POST, IBL
  - NOR: BIOS MCSDK Demo
  - NAND: Linux MCSDK Demo
  - EEPROM/Flash writers
- CCS 5.0
  - IDE
  - C667x EVM GEL/XML files
- BIOS MCSDK 2.0
  - Source/binary packages
- Linux MCSDK 2.0
  - Source/binary packages

EVM Flash Contents

- EEPROM 128 KB
  - POST
  - IBL
- NOR 16 MB
  - BIOS MCSDK "Out of Box" Demo
- NAND 64 MB
  - Linux MCSDK Demo

Online Collateral

TMS320C667x processor website
http://focus.ti.com/docs/prod/folders/print/tms320c6678.html
http://focus.ti.com/docs/prod/folders/print/tms320c6670.html

MCSDK website for updates
http://focus.ti.com/docs/toolsw/folders/print/bioslinuxmcsdk.html

CCS v5

Developer's website
Linux: http://linux-c6x.org/
For More Information

• MCSDK Product Folder:
  http://www.ti.com/tool/bioslinuxmcsdk

• Download CCSv5 and MCSDK software:

• KeyStone C66x Multicore Wiki Resources

• For questions regarding topics covered in this training, visit the support forums at the TI E2E Community website http://e2e.ti.com