PDK (Platform Development Kit) Getting Started

Automotive Processors

Agenda

- PDK Overview
- PDK Software Architecture
- PDK Directory structure
- PDK Pre-requisite and Build instructions
- Running Examples
- Important Links

PDK Overview (1/2)

- Platform Development Kit (PDK) is a single scalable software driver package to offer streamlined development across different TDA processors
- Provides peripheral programming interface - Easy to use SW interface
- Provides software portability across devices for a given peripheral
- Provides stand-alone examples to demonstrate the capabilities of the peripherals
- Product is designed to scale for each SoC. Single package is used to support more than one SoC. The product packaging/folder takes care of scalability for new IPs of new SoCs.
- MISRA-C compliant

PDK Overview (2/2)

- All required libraries and drivers provided are OS agnostic with few additional TI-RTOS drivers.
- Can be used stand-alone
- Not a middleware complete solution
- Supports following devices
  - TDA2xx
  - TDA2Ex
  - TDA3xx
  - TDA2Px

PDK Software Architecture

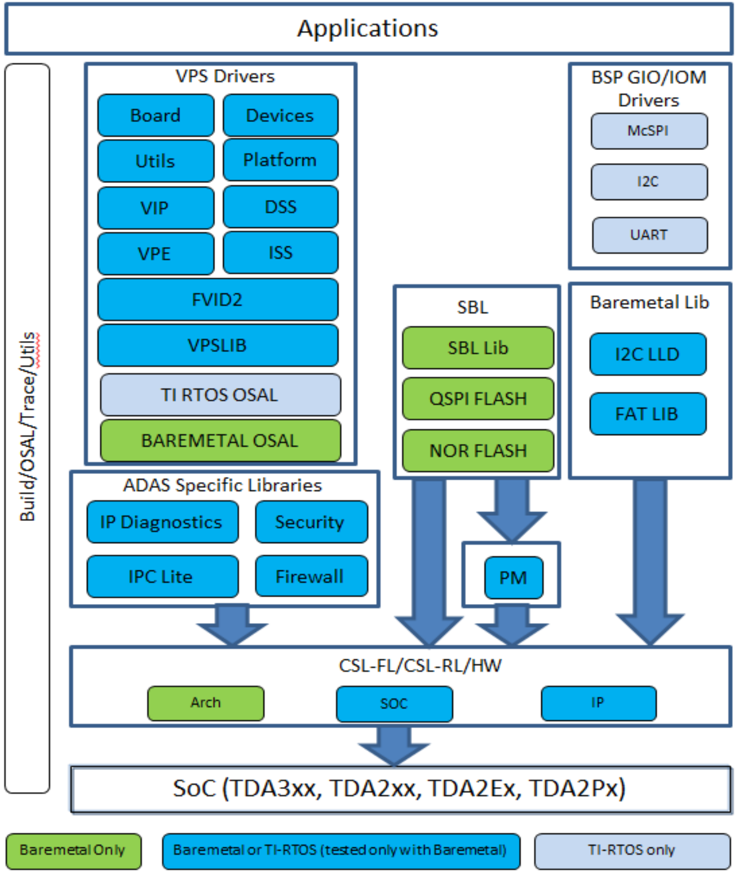
CSL Libraries: **UART, I2C GPIO, Mailbox, Spinlock, EDMA, GPMC, McASP, McSPI, OCMC, QSPI, SBL, MMU, Timer, MMCSD, WDTIMER, PCIe**

Driver libraries: I2C, QSPI, FAT, NOR, VPS (VIP/DSS/VPE)

SD, NOR and QSPI bootmodes

UART console utility

PDK Software Architecture



Applications

Build/OSAL/Trace/Utils

VPS Drivers

Board | Devices | Utils | Platform | VIP | DSS | VPE | ISS | FVID2 | VPSLIB | TI RTOS OSAL | BAREMETAL OSAL

BSP GIO/IOM Drivers

McSPI | I2C | UART

SBL

SBL Lib | QSPI FLASH | NOR FLASH

Baremetal Lib

I2C LLD | FAT LIB

ADAS Specific Libraries

IP Diagnostics | Security | IPC Lite | Firewall

PM

CSL-FL/CSL-RL/HW

Arch | SOC | IP

SoC (TDA3xx, TDA2xx, TDA2Ex, TDA2Px)

Baremetal Only

Baremetal or TI-RTOS (tested only with Baremetal)

TI-RTOS only

Software Modules – CSL Module

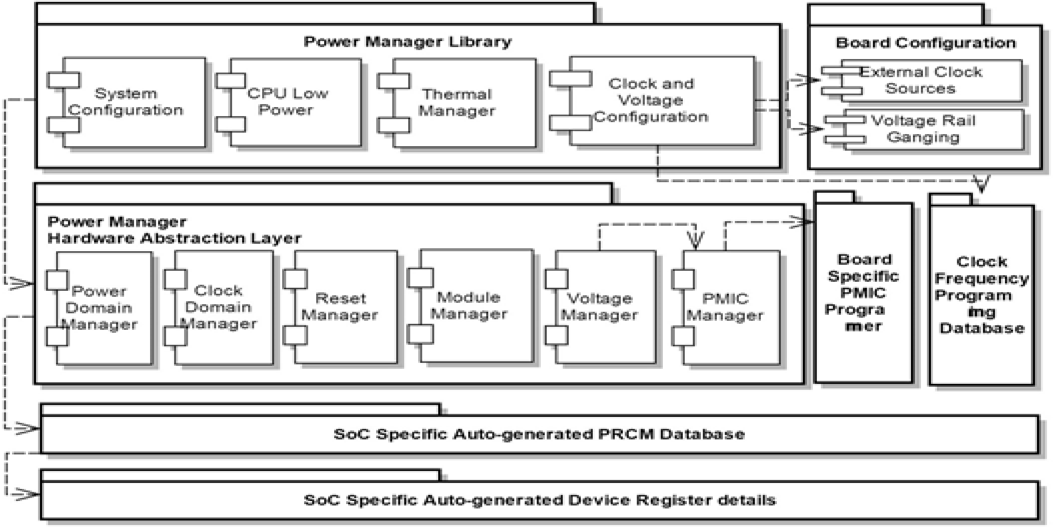
- Chip Support Library contains
  - APIs to configure and access peripherals
  - Simple, consistent and intuitive to use APIs
  - Logical group of register reads and writes to get a functional layer for a given IP
  - Stateless functions; blocking and runs to completion
  - System Configuration Code:
    - Provides functions specific to SoC level System Configuration.
    - Provides the following functionalities:
      - Start-up Code
      - Interrupt-vector initialization
      - Low level CPU specific code
    - May involve assembly code and hence can be tools specific
  - Sub module to further abstract and group the functionality.

Software Modules – CSL Module

Sub Modules	Description
Arch	This sub module contains the core specific functionality and initialization code for the system and cores. The interrupt controller API's to configure and enable/disable the interrupt is also part of this sub module. Source code is available under 'csl/arch' folder.
SoC	SoC specific defines are abstraction under this sub module. SoC specific base address, macros and cross bar configuration code is available under this sub module. Source code is available under 'csl/soc' folder.
IP Specific Code	API to use the functionality of different IP's is available as part of this sub module. Source code is available under 'csl/src/ip' folder.

Software Modules – PM

- PM Module
  - Provides API to configure Power domain and Clock domain
  - Set power state for different modules
  - Set clock rate for different module
  - Supports Dynamic Power Management
  - Supports Software Thermal Management


Power Manager Library
System Configuration
CPU Low Power
Thermal Manager
Clock and Voltage Configuration
Board Configuration
External Clock Sources
Voltage Rail Ganging
Power Manager Hardware Abstraction Layer
Power Domain Manager
Clock Domain Manager
Reset Manager
Module Manager
Voltage Manager
PMIC Manager
Board Specific PMIC Progra mer
Clock Frequency Program ing Database
SoC Specific Auto-generated PRCM Database
SoC Specific Auto-generated Device Register details

Power Management Software Components

Software Modules – PM

- PM Module types
  - Master Standby
    - Valid for Initiators to the Interconnect
    - When Master does not want clocks configure IP level SYSCONFIG MIDLEMODE or STANDBYMODE
  - Slave Idle
    - Valid for modules which respond to requests.
    - Configure PRCM register CLKCTRL. MODULEMODE.
    - Configure IP level SYSCONFIG SIDLEMODE or IDLEMODE
- PM Module can be further split into
  - PRCM Database: Abstracts the SOC specific PRCM details.
  - PMLIB: Library which can be used by application to initialize the system
  - PMHAL: Contains board specific PMIC functions

Software Modules – SBL

- SBL
  - Initializes the execution environment for multi-core application
  - Configure the Voltage rails (AVS Class 0)
  - Initialize the Interrupt Controller
  - Setup the ADPLL values
  - Power on I/O peripherals
  - Power on slave cores
  - Configure required PADs
  - Initialize DDR
  - Load the multicore application image into DDR
  - Bring slave cores out of Reset
  - Boot Modes: QSPI, QSPI_SD, NOR and MMCSD application images

Software Modules – ADAS Libraries

- Diagnostic Library
  - Contains set of software function to diagnosing the health of various modules and interfaces of the device.
  - Supports ECC, CRC, ADC, CAN and SPI modules.
  - Initializes the modules the verify the functionality in different configurations

- IPC Lite Library
  - OS agnostic library to provide two way communication between hosts
  - Built on top of mailbox
  - Supports 32bit or 64bit value to be transferred
  - Supports event multiplexing

Software Modules – ADAS Libraries

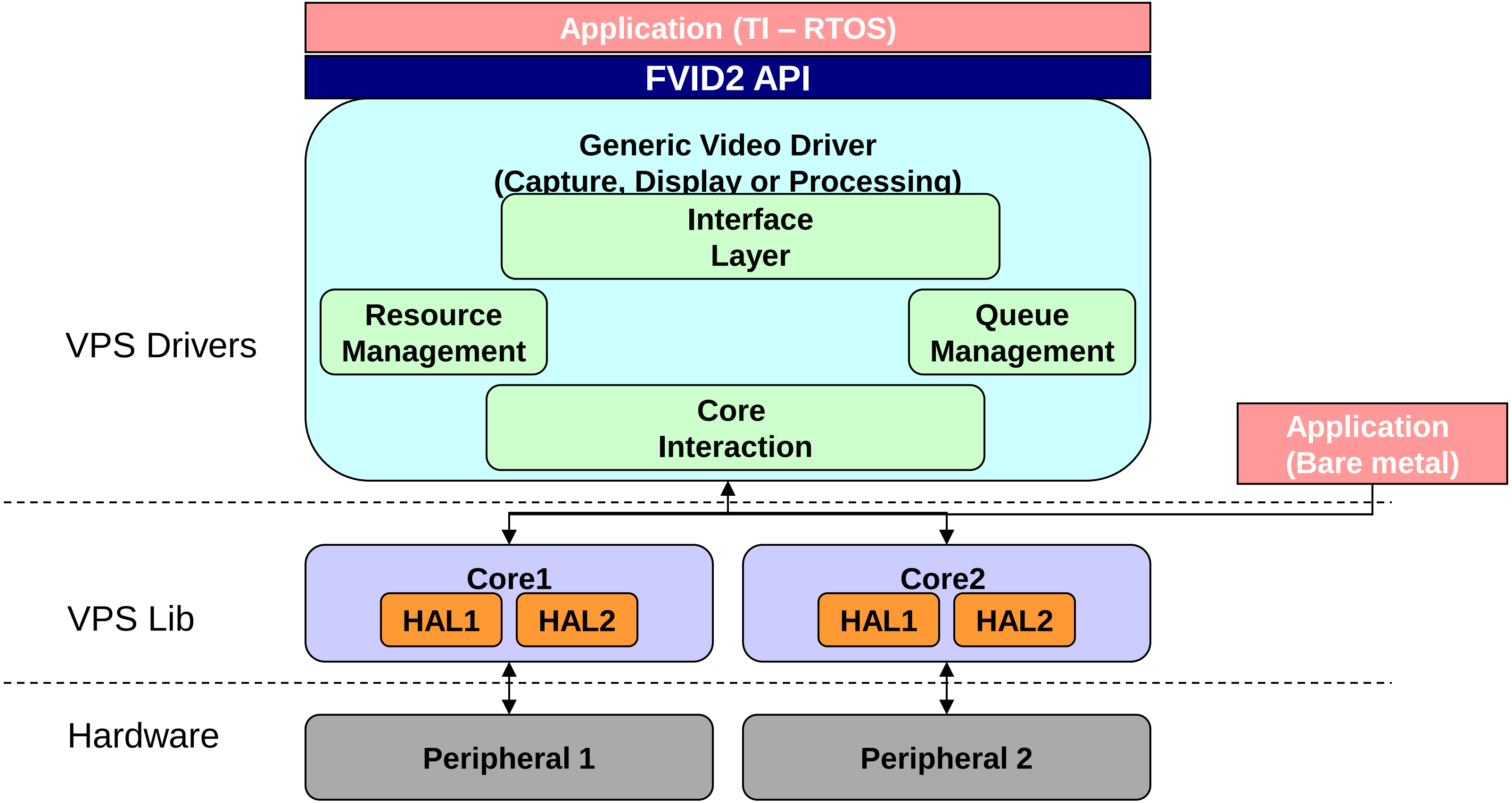
- Firewall
  - Provides set of function to configure the system to prevent unauthorized access to the modules
  - Supports L3 and L4 interconnect

- Security
  - Provides no-OS platform support for security specific features
  - Required to work with TDA3xx GP prime variant of devices
  - Provide code for AES128-CBC and SHA256 encryptions

Software Modules – VPS

- VPS Module
  - Provides set of API’s to allow users to make use of all video hardware features.
  - VPS drivers are based on FVID2 interface
  - Platform interface: SoC specific functionality
  - Board interface: Board-specific functionality
  - Decoder/Encoder interfaces: Interfaces to use different decoders or encoders
- VPS consists of following modules
  - Video Input Port (VIP)
  - Video Processing Engine (VPE)
  - Display Sub System (DSS)
  - Imaging sub system (ISS) includes CAL, ISP, SIMCOP

Software Partition – VPS Diver Stack



Software Modules – BSP LLD

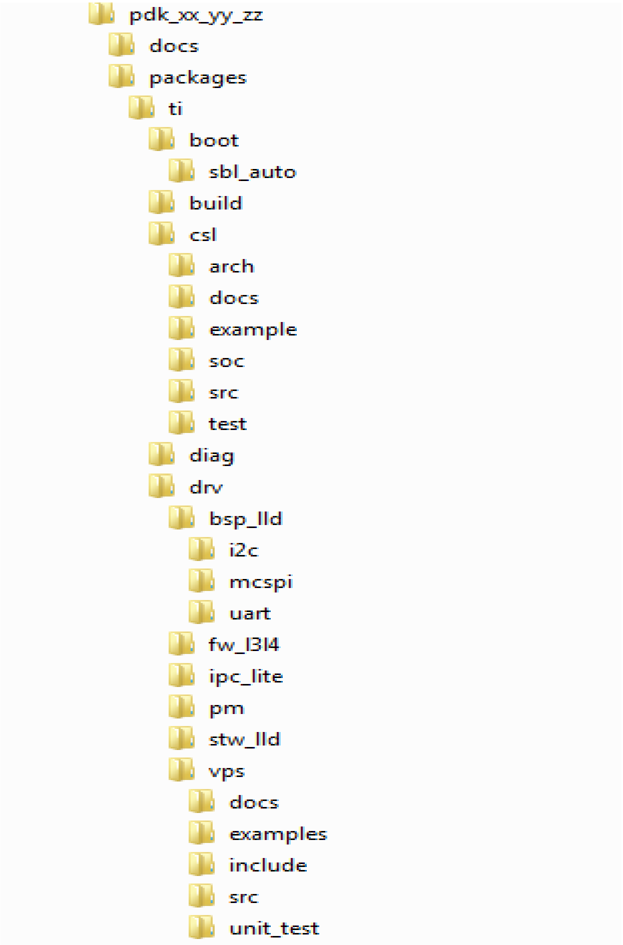
- Drives based on the sysbios operating system.
- Implemented in the GIO models of the sysbios.
- Polled, interrupt and DMA mode of transfers are supported.
- Supports master and slave mode for McSPI and I2C
- Supports
  - I2C:
  - UART:
  - McSPI:

Software Modules – Examples, Utility and tools

Tools/Utilities	Description
mFlash	mFlash tool is used provide a way to flash the binary images to QSPI Flash memory using the UART interface. Tool is available @ “packages/ti/boot/sbl_auto/tools/mflash”
Multicore Image generator	These are the scripts provided bmFlash tool is used provide a way to flash the binary images to QSPI Flash memory using the UART interface. Tool is available @ “packages/ti/boot/sbl_auto/tools/mflash” y the TI to generate the multicore app images using individual core binaries. Tool is available @ “packages/ti/boot/sbl_auto\tools/MulticoreImageGen_xxxx.bat”
QSPI/Nor flash writer	QSPI or NOR flash writer are required to flash the SBL AppImage in QSPI/NOR flash. Tool is available @ Packages/ti/boot/sbl_auto/qspiflash/examples Packages/ti/boot/sbl_auto/norflash/examples
UART console	Standard API to configure, read and write data on the UART interface. Packages/ti/csl/example/utils/uart_console

Directory Structure

- Uses simple and intuitive directory structure
- Provides scope to accommodate new SoCs in the same directory structure
- Build folder contains all the make rules for building PDK
- Each module/library has Include folder containing the corresponding header files
- Drivers and libraries are separated from low level driver, categorized and under single directory.
- Examples and sample applications are part of corresponding module



```
pdk_xx_yy_zz
  docs
  packages
    ti
      boot
        sbl_auto
      build
      csl
        arch
        docs
        example
        soc
        src
        test
      diag
      drv
        bsp_lld
          i2c
          mcspi
          uart
        fw_l3l4
        ipc_lite
        pm
        stw_lld
        vps
          docs
          examples
          include
          src
          unit_test
```

PDK Pre-requisite

- Code Composer Studio (v 5.4.0.00091 or above)
- Compiler and Tool chains
  - TMS470
  - TI Compiler C6000
  - TI Compiler ARP32
  - Linaro GCC tool chain for A15
  - BIOS/XDC
  - EDMA LLD
- Flash tools:
  - NOR Flash Writer
  - QSPI Flash Writer

- Development Board
- PDK package
  - PDK package is available as part of Processor SDK.
  - PDK package will be available at "vision_sdk_xx_yy_zz/ti_components/drivers/pdk"
- Emulator
  - XDS560v2
  - XDS200
- SD Card

PDK Build

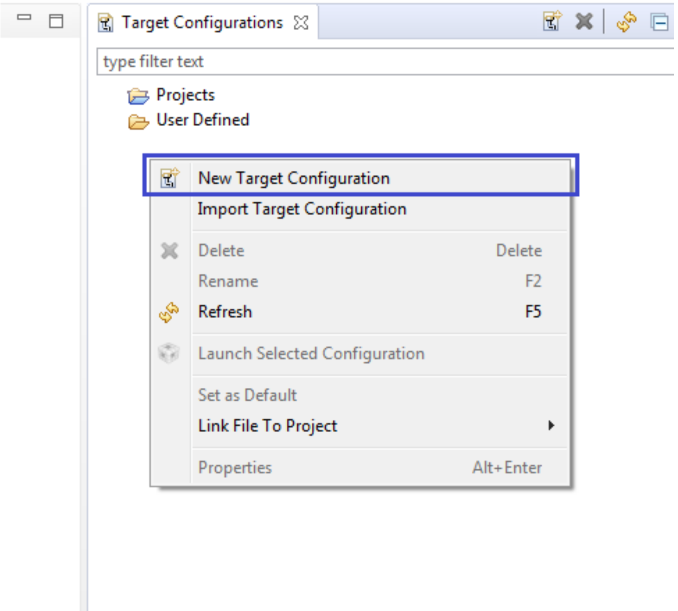
- Supports Makefile based build
- PDK will be automatically build while building Vision SDK. PDK can be build independently also and command's the should be issued from "<pdk>/packages/ti/build" folder
- Following steps application for building PDK independently
  - Makerules for different cores are specified in packages/ti/build/makerules
  - Environment variables are specified in packages/ti/build/makerules/env.mk
  - *gmake allcore BOARD=<BOARD> builds all the components*
  - *gmake <example/module> BOARD=<BOARD> builds specific module or example*

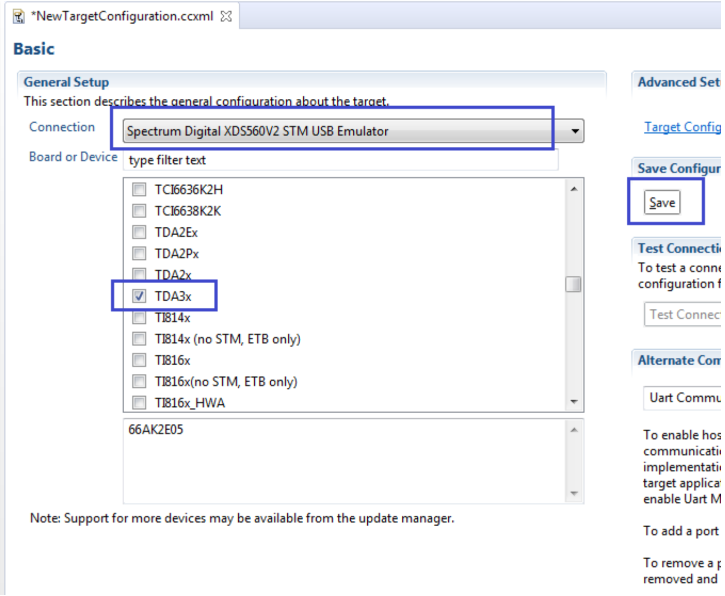
<BOARD> can be tda3xx-evm, tda2xx-evm, tda2ex-evm or tda2px-evm

Refer to http://processors.wiki.ti.com/index.php/PDK/PDK_TDA_Building_The_Package

Running Examples (1/2)

Examples can be run either in Debug mode using CCS or SD boot mode using SD card or QSPI/NOR boot mode using on-board flash. Refer to SBL user-guide for more details about flashing SBL, appimage, preparing SD etc.





This section describes the general configuration about the target.

Connection: Spectrum Digital XDS560V2 STM USB Emulator

Board or Device: type filter text

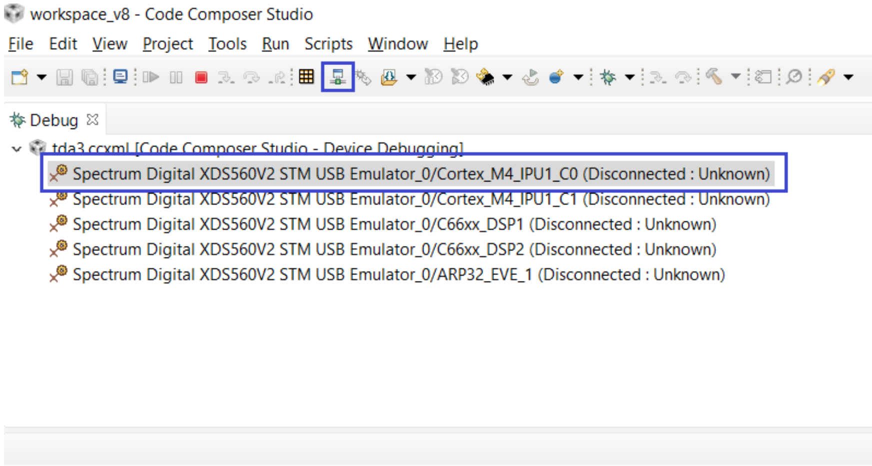
- ☐ TCI6636K2H
- ☐ TCI6638K2K
- ☐ TDA2Ex
- ☐ TDA2Px
- ☐ TDA2x
- ☑ TDA3x
- ☐ TI814x
- ☐ TI814x (no STM, ETB only)
- ☐ TI816x
- ☐ TI816x(no STM, ETB only)
- ☐ TI816x_HWA

66AK2E05

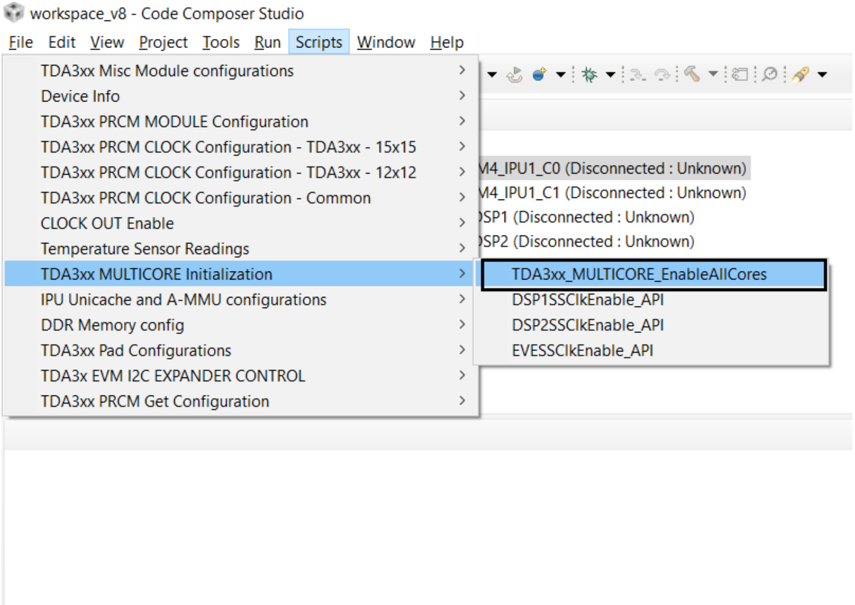
Note: Support for more devices may be available from the update manager.

Running Examples (2/2)

Examples can be run either in Debug mode using CCS or SD boot mode using SD card or QSPI/NOR boot mode using on-board flash. Refer to SBL user-guide for more details about flashing SBL, appimage, preparing SD etc.


workspace_v8 - Code Composer Studio
File Edit View Project Tools Run Scripts Window Help
Debug

tda3.ccxml [Code Composer Studio - Device Debugging]
- Spectrum Digital XDS560V2 STM USB Emulator_0/Cortex_M4_IPU1_C0 (Disconnected : Unknown)
- Spectrum Digital XDS560V2 STM USB Emulator_0/Cortex_M4_IPU1_C1 (Disconnected : Unknown)
- Spectrum Digital XDS560V2 STM USB Emulator_0/C66xx_DSP1 (Disconnected : Unknown)
- Spectrum Digital XDS560V2 STM USB Emulator_0/C66xx_DSP2 (Disconnected : Unknown)
- Spectrum Digital XDS560V2 STM USB Emulator_0/ARP32_EVE_1 (Disconnected : Unknown)


workspace_v8 - Code Composer Studio
File Edit View Project Tools Run Scripts Window Help
TDA3xx Misc Module configurations
Device Info
TDA3xx PRCM MODULE Configuration
TDA3xx PRCM CLOCK Configuration - TDA3xx - 15x15
TDA3xx PRCM CLOCK Configuration - TDA3xx - 12x12
TDA3xx PRCM CLOCK Configuration - Common
CLOCK OUT Enable
Temperature Sensor Readings
TDA3xx MULTICORE Initialization
IPU Unicache and A-MMU configurations
DDR Memory config
TDA3xx Pad Configurations
TDA3x EVM I2C EXPANDER CONTROL
TDA3xx PRCM Get Configuration
TDA3xx_MULTICORE_EnableAllCores
DSP1SSClkEnable_API
DSP2SSClkEnable_API
EVESSClkEnable_API
M4_IPU1_C0 (Disconnected : Unknown)
M4_IPU1_C1 (Disconnected : Unknown)
SP1 (Disconnected : Unknown)
SP2 (Disconnected : Unknown)

Important links

- PDK Software Developer Guide: http://processors.wiki.ti.com/index.php/PDK/PDK_TDA_Software_Developer_Guide
- STW/BSP to PDK Migration Guide: http://processors.wiki.ti.com/index.php/PDK/BSP_STW_To_PDK_Migration_Guide
- VPS User Guide: http://processors.wiki.ti.com/index.php/PDK/PDK_TDA_VPS_User_Guide
- VPS training: Available at <PDK>/packages/ti/drv/vps/docs/training/
- PM User Guide: http://processors.wiki.ti.com/index.php/PDK/PDK_TDA_PM_User_Guide
- SBL User Guide: http://processors.wiki.ti.com/images/b/b7/TDA_SBL_UserGuide.pdf
- Application Notes: http://processors.wiki.ti.com/index.php/PDK/PDK_TDA_Application_notes