NDK, PA Resource Wiki for Keystone Devices

This wiki is under construction

Contents

Network Development Kit (NDK) and Packet Accelerator (PA) Resource Wiki for Keystone Devices

Overview

This wiki article is a collection of frequently asked questions (FAQ) on NDK and PA of Keystone family of devices; along with some useful documents and software links.

FAQ on NDK and PA of Keystone

Q When NDK's Helloworld or Client example is run on other than core0, it gives the error on CCS console. Is it possible to run on cores other than core0?

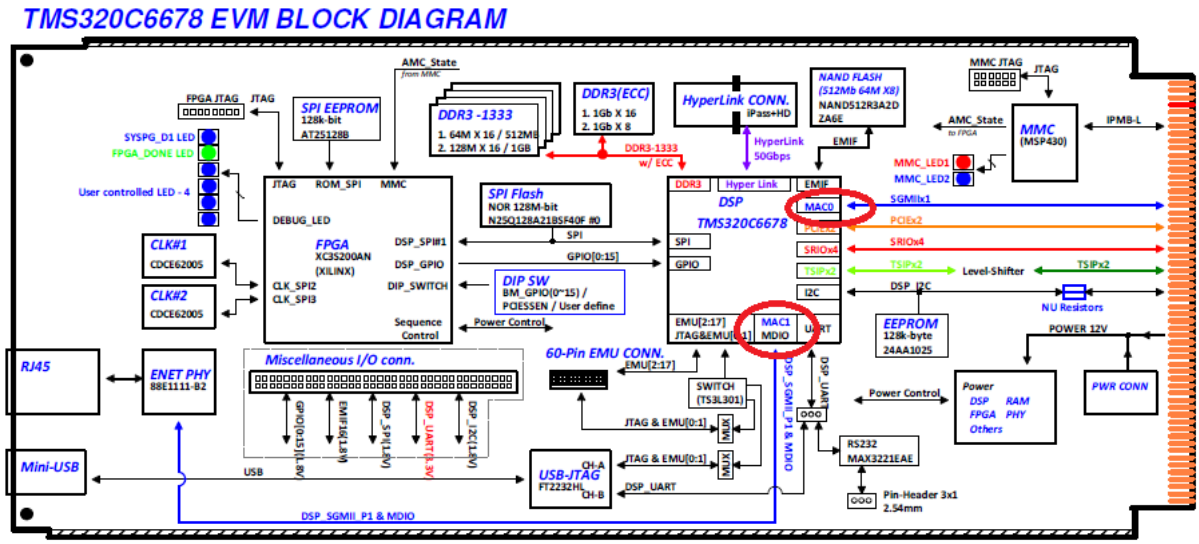
The NDK and NIMU based examples are designed to run on core0 by default. Yes, it is possible to load and run on the other core like core1, core2. But the NIMU driver and example code needs to be modified according to core on which it runs. There is no solution for running simultaneous NDK stacks on multiple cores. The attached code is based on "**mcsdk_2_01_02_06**" release.

Please refer to the attached code below.

File:NDK-Core1.zip

Q Where and how to enable the EMAC0 on C6678 EVM's AMC connector?

C6678 processor has support for two EMAC interfaces, one of which EMAC1 is connected to Gigabit RJ-45 connector of Shannon C6678 EVM whereas EMAC0 had been connected to AMC connector of Shannon C6678 EVM.



By default, NIMU driver does not have a support for EMAC0 on AMC connector, TI has provided the files for enabling EMAC0 on EVM's AMC connector in the following attachment.

Please note that the current NIMU driver doesn't support simultaneous NDK support on both EMAC ports.The attached code is based on "**mcsdk_2_01_02_06**" release.

Also, EMAC0 can be used for in MAC to MAC mode using Dual EVM Breakout card between two Shannon EVM boards.

http://processors.wiki.ti.com/images/0/00/DualEVM_BOC_quick_setup_guide_09SEPT13.pdf

File:NIMU-EMAC0.zip

Q Can I find the network throughput performance code in the MCSDK?

Yes, the network throughput performance can be resulted through running the HUA demo. Refer the below E2E to find the details to improve upon the TCP throughput.

http://e2e.ti.com/support/dsp/c6000_multi-core_dsps/f/639/p/207175/743100.aspx#743100

Q The PA EMAC example is worked on core0; Will this example work on other cores?

Yes, the PA emacExample is designed to run on core 0. However, it is pretty easy to make it run on other cores as a single core application which means that it owns the entire NETCP.
There is one-to-one corresponding between the accChannelNum to the interrupt event of each core where queue number does not really matter.
The accChannelNum should be equal to the core number.

You can simply replace *PA_ACC_CHANNEL_NUM* with *PA_ACC_CHANNEL_NUM* (core) as the following definition:
#define PA_ACC_CHANNEL_NUM (core) (core)

Q I have seen PA EMAC example, this code is designed with loopback modes. What is the difference between external loopback and internal loopback?

Refer to the README file in the same example folder that explains most of the configuration.
The PA examples are having MACROs to support multiple platforms including CCS simulator and various configurations.

CPSW_LOOPBACK_INTERNAL (1): SGMII internal loopback, i.e. the SGMII will loopback the packet as it is.
CPSW_LOOPBACK_EXTERNAL (2): The SGMII internal loopback is turned off. However, the application expects the external device (PC, Phy loopback) will loopback the packet as it is.

Q NDK client example is not working when application reloading/restarting?

You have to modify your application(s) to reset resources/peripherals needed to restart the NDK stack.
Find the NDK based client example restart source available at:

http://e2e.ti.com/support/dsp/c6000_multi-core_dsps/f/639/p/247150/921455#921455

Q Will the NIMU driver uses the PA? Is there any release combined NIMU and PA?

No. This NIMU layer calls underlying hardware driver/CSL. In case of keystone devices, the NIMU code does not use PA. It bypasses PA and sends packet directly to the QMSS Queue648.

http://processors.wiki.ti.com/index.php/BIOS_MCSDK_2.0_User_Guide#Network_Interface_Management_Unit_.28NIMU.29_Driver

TI provides both NDK and PA LLD. The application can either use NDK or PA LLD with a network stack provided by the customer himself. If you choose to use NDK, you application should interface with NDK only, i.e. invoking NDK APIs for all data traffic. If you choose to use PA LLD, you need to write your own network stack to interface with PA LLD and other low layer software stacks such as CPPI and QMSS LLDs.

The NDK itself is device-independent. It uses the device-specific NIMU to interface with low-level device driver. In the current implementation, the TCI6678 NIMU only uses PASS to perform device MAC address filtering so that only broadcast, multicast and device-specific MAC packet will be delivered to NDK. In the egress direction, the TX packets are pushed to the CPSW queue (Q#648) directly. It is up to the platform team to determine how much functionality of PASS that NIMU will take advantage of in the future. On keystone devices, the NIMU layer is interface between NDK stack and the NETCP. It does not utilize PA subsystem currently. There is no release combined with the NIMU and PA.

Q Is there updated NIMU driver support for fast and reliable TCP with NDK on keystone devices?

This patched nimu.c file is tried and it works on Rev.2A TMDSEVM6678L boards reliably.

Please refer to the attached code.

File:Nimu eth.zip

Q How to enable Jumbo packet support for C6678 ?

Refer to the following TI wiki page.

http://processors.wiki.ti.com/index.php/Enabling_Jumbo_Packet_Support_for_C6678

Q How to run the NDK examples on Keystone devices ?

Refer to the following TI wiki page.

http://processors.wiki.ti.com/index.php/Running_NDK_examples_for_Keystone_devices

{{	Keystone=	C2000=*For*	DaVinci=*For*	MSP430=*For*	OMAP35x=*For*	OMAPL1=*For*	MAVRK=*For*	*For techni*
1. switchcategory:MultiCore=	▪ For technical support on	*technical support on*	*technical support on*	*technical support on*	*technical support on*	*technical support on*	*technical support on*	*please po questions*

- For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum
- For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum

Please post only comments related to the article **NDK, PA Resource Wiki for Keystone Devices** here.

MultiCore devices, please post your questions in the C6000 MultiCore Forum

- For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum

Please post only comments related to the article **NDK, PA Resource Wiki for Keystone Devices** here.

*the C2000 please post your questions on The C2000 Forum. Please post only comments about the article **NDK, PA Resource Wiki for Keystone Devices** here.*

*DaVincoplease post your questions on The DaVinci Forum. Please post only comments about the article **NDK, PA Resource Wiki for Keystone Devices** here.*

*MSP430 please post your questions on The MSP430 Forum. Please post only comments about the article **NDK, PA Resource Wiki for Keystone Devices** here.*

*OMAP please post your questions on The OMAP Forum. Please post only comments about the article **NDK, PA Resource Wiki for Keystone Devices** here.*

*OMAP please post your questions on The OMAP Forum. Please post only comments about the article **NDK, PA Resource Wiki for Keystone Devices** here.*

*MAVRK please post your questions on The MAVRK Toolbox Forum. Please post only comments about the article **NDK, PA Resource Wiki for Keystone Devices** here.*

*http://e2e. Please po comments article **ND Resource Keystone** here.*

}}

Links

Amplifiers & Linear
Audio
Broadband RF/IF & Digital Radio
Clocks & Timers
Data Converters
DLP & MEMS
High-Reliability
Interface
Logic
Power Management
Processors

- ARM Processors
- Digital Signal Processors (DSP)
- Microcontrollers (MCU)
- OMAP Applications Processors

Switches & Multiplexers
Temperature Sensors & Control ICs
Wireless Connectivity