Ambarella

SDK6 Release Note

A12 SDK Version 6.2.002

Version 1.1

July 8, 2015

Ambarella

Confidentiality Notice:

The contents of this document are proprietary and confidential information of Ambarella Inc.

The material in this document is for information only. Ambarella assumes no responsibility for errors or omissions and reserves the right to change, without notice, product specifications, operating characteristics, packaging, ordering, etc. Ambarella assumes no liability for damage resulting from the use of information contained in this document. All brands, product names and company names are trademarks of their respective owners.

US
3101 Jay Street
Ste.110
Santa Clara, CA 95054, USA
Phone: +1.408.734.8888
Fax: +1.408.734.0788

Hong Kong
Unit A&B, 18/F, Spectrum Tower
53 Hung To Road
Kwun Tong, Kowloon
Phone: +85.2.2806.8711
Fax: +85.2.2806.8722

Korea
6 Floor, Hanwon-Bldg.
Sunae-Dong, 6-1, Bundang-Gu
SeongNam-City, Kyunggi-Do
Republic of Korea 463-825
Phone: +031.717.2780
Fax: +031.717.2782

China - Shanghai
9th Floor, Park Center
1088 Fangdian Road, Pudong New District
Shanghai 201204, China
Phone: +86.21.6088.0608
Fax: +86.21.6088.0366

Taiwan
Suite C1, No. 1, Li-Hsin Road 1
Science-Based Industrial Park
Hsinchu 30078, Taiwan
Phone: +886.3.666.8828
Fax: +886.3.666.1282

Japan - Yokohama
Shin-Yokohama Business Center Bldg. 5th Floor
3-2-6 Shin-Yokohama, Kohoku-ku,
Yokohama, Kanagawa, 222-0033, Japan
Phone: +81.45.548.6150
Fax: +81.45.548.6151

China - Shenzhen
Unit E, 5th Floor
No. 2 Finance Base
8 Ke Fa Road
Shenzhen, 518057, China
Phone: +86.755.3301.0366
Fax: +86.755.3301.0966

I Contents

II Preface

This document provides technical details using a set of consistent typographical conventions to help the user differentiate key concepts at a glance.

Conventions include:

Example	Description
AmbaGuiGen, **DirectUSB** **Save**, **File > Save** **Power**, **Reset**, **Home**	Software names GUI commands and command sequences Computer / Hardware buttons
Flash_IO_control **da**, **status**, **enable**	Register names and register fields. For example, **Flash_IO_control** is the register for global control of Flash I/O, and bit 17 (**da)** is used for DMA acknowledgement.
GPIO81, **CLK_AU**	Hardware external pins
VIL, VIH, VOL, VOH	Hardware pin parameters
INT_O, RXDATA_I	Hardware pin signals
amb_performance_t ***amb_operating_mode_t*** **amb_set_operating_mode()**	API details (e.g., functions, structures, and type definitions)

`/usr/local/bin`
`success = amb_set_operating_mode (amb_base_address, & operating_mode)`
	User entries into software dialogues and GUI windows
	File names and paths
	Command line scripting and Code

Table II-1. Typographical Conventions for Technical Documents.

Additional Ambarella typographical conventions include:

- Acronyms are given in UPPER CASE using the default font (e.g., AHB and DDRIO).
- Names of Ambarella documents and publicly available standards, specifications, and databooks appear in *italic* type.

1 Overview

1.1 Overview: Introduction

The Ambarella team is pleased to announce the release of version 6.2.002 of the A12 Software Development Kit (SDK).

This release includes the **Middleware Support Package (MSP)**, a robust framework which allows customers to fully utilize the functionality of the A12 SoC in a straightforward manner. The MSP provides flexible bitstream management which can support a variety of complex applications, such as multiple stream storage, networking, and more. The version 6.2.002 release also provides an updated **Connected App**, an application designed to demonstrate various features that can be implemented with Ambarella API libraries.

The A12 Middleware Support Package and Connected App currently support the following hardware platforms:

1. The Dragonfly Evaluation Kit (EVK) Reference Board (Figure 1-1)
2. The Taroko EVK Reference Board (Figure 1-2)

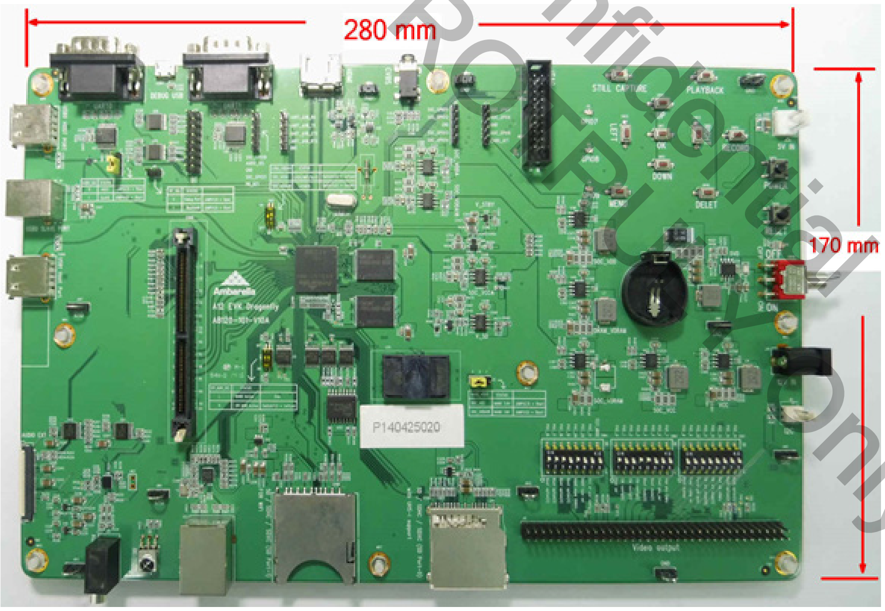

Figure 1-1. A12 SDK: Dragonfly EVK Board.

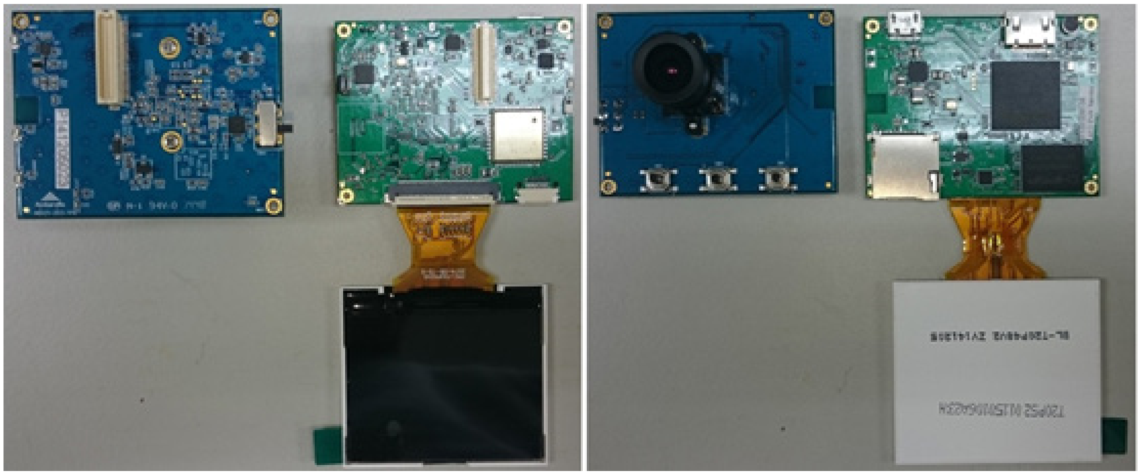

Figure 1-2. A12 SDK: Taroko EVK Board.

2 A12 SDK Version 6.2.002

2.1 SDK 6.2.002: Introduction

The Ambarella A12 SDK enables the development of high-performance camera products in the sports, wearable (consumer as well as police/security) and automotive market segments. The SDK provides the necessary software and hardware tools to enable designers to create customized, fully-featured camera products.

Version 6.2.002 of the A12 SDK includes the following software modules:

1. **Middleware Support Package:**
   - ThreadX
     - Middleware Unit Test (Source)
     - Middleware/Codec (Library)
     - Middleware/FIFO (Library)
     - Middleware/Data (Library)
   - Linux
     - Network Apps (Library)
2. **Connected App:**
   - ThreadX
     - Connected App (Source)
     - AppLib (Library)

This document is organized as shown below.

- (Section 2.2) SDK 6.2.002: Middleware Support Package
- (Section 2.3) SDK 6.2.002: Connected App
- (Section 2.4) SDK 6.2.002: SDK 6.2.002: Known Issues
- (Section 2.5) SDK 6.2.002: QA Test Results
- (Section 2.6) SDK 6.2.002: Power Measurement Test Results

For a summary of differences between SDK version 6.2.002 and the previous SDK version, refer to Appendix 1.

2.1.1 Introduction: Diagram

Figure 2-1 below provides a software block diagram of the A12 SDK version 6.2.002.

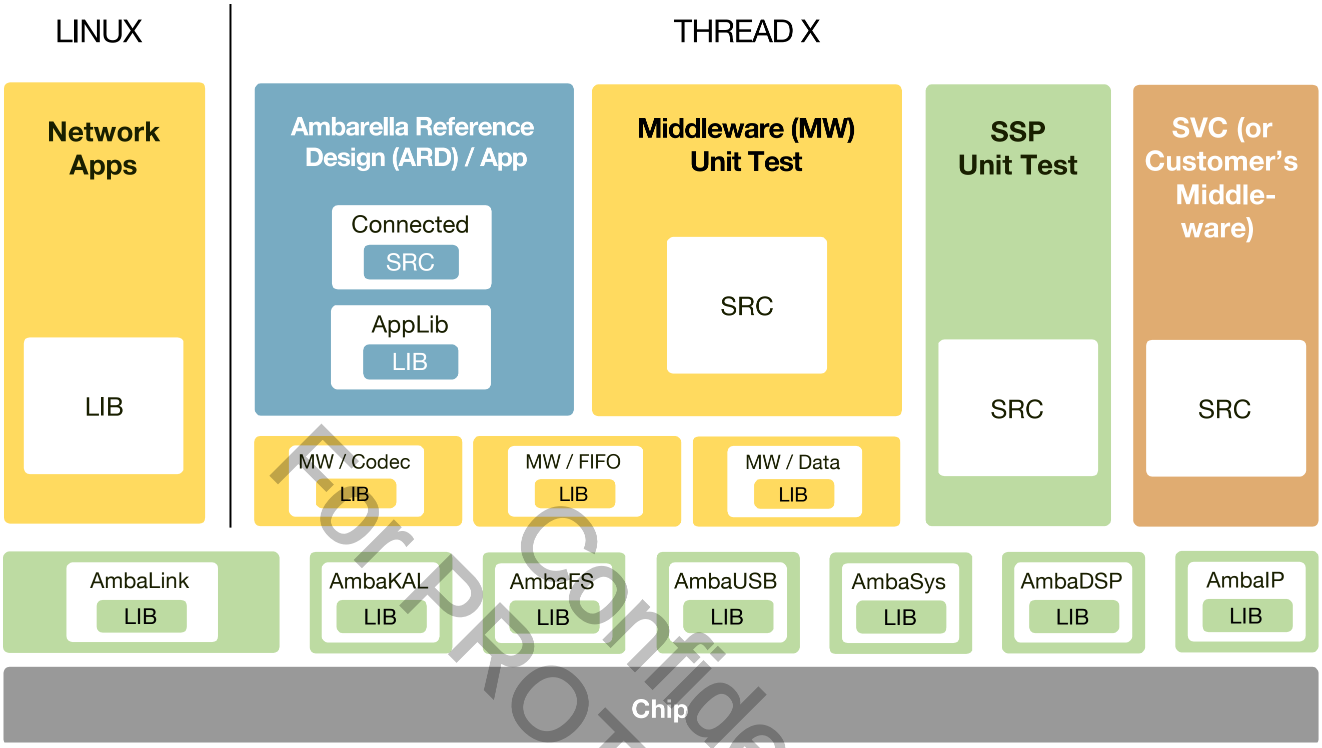

Figure 2-1. A12 SDK: Software Block Diagram.

2.1.2 Introduction: SDK Architecture

The A12 SDK consists of three distinct layers:

- (Section 2.1.2.1) System Support Package (SSP)
- (Section 2.1.2.2) Middleware Support Package (MSP)
- (Section 2.1.2.3) Ambarella Reference Design (ARD)

2.1.2.1 System Support Package (SSP)

The A12 System Support Package (SSP) provides the fundamental services used to run Ambarella chips. This package includes basic unit tests and system service code (sample muxer/demuxers, simple GUIs, file-naming rules, etc.).

Of the available A12 SDK packages, the SSP enables the greatest degree of control and customization when developing new products. Customers who select the SSP package are typically competing in markets where innovation and diversification are key requirements for success. In addition, because the use of the SSP package often leads to a fuller understanding of our technologies, customers who select this package are typically engaged in longer-term, multi-product relationships with Ambarella.

It should be noted, however, that **leveraging the capabilities of the SSP requires the investment of considerable engineering resources on the part of the customer**.

For example, developing products with the SSP requires (a.) studying the source code for Unit Tests/System Service Code, (b.) understanding fundamental SSP behaviors and protocols, and (c.) developing customized software from the ground up, especially in the case of a first-time product (Ambarella typically maintains similar API interfaces across generations of SoCs; therefore, the development cycles for future products may be shortened).

The SSP layer consists of the components listed below.

- **System Libraries:**
  - AmbaKAL: RTOS Kernel Abstraction Layer
  - AmbaFS: File System
  - AmbaUSB: USB Stack
  - AmbaSys: System I/O drivers based on Ambarella chip design
- **AmbaDSP: DSP Support Package**
  - Used to control the DSP inside Ambarella chips
- **AmbaLink**: Provides RTOS and Linux communication and network support
- **AmbaIP**: Ambarella reference AE/AWB/ADJ libraries
- **SSP Unit Test**: Unit tests for the SSP layer
- **SVC (System serVice Codes)**: Sample application over SSP

In a typical SSP release, customers will receive the following:

1. **SSP Libraries**
2. **SSP Unit Test Source Code**
3. **SVC Source Code**
4. **Common Service Source Code**: Frequently-used small utilities
5. **Image Quality Utility Libraries**: Utilities for calibration, AE/AWB/ADJ scheduling, bitrate monitoring

2.1.2.2 Middleware Support Package (MSP)

The Middleware Support Package (MSP) enables the full utilization of SSP capabilities via easy-to-control mechanisms, allowing customers to pursue a straightforward product development path.

Because a majority of SSP protocols are either managed or translated to simplified forms, the MSP does not require customers to understand low-level SSP protocols. Customers can create diverse features or refine existing features (demonstrated in the relevant application) from the middleware level.

It should be noted, however, that **leveraging the capabilities of the MSP requires the investment of engineering resources on the part of the customer**. Depending upon customer goals, this investment can be either:

1. **Light**: The customer studies the ARD/APP Applib source code to gain an understanding of how the Applib utilizes the middleware layer to implement a specific feature.
2. **Heavy**: The customer studies the Unit Test source code in order to learn how to use middleware APIs directly.

In either case, customers who select the MSP package will be limited by the middleware architecture and available feature set. For this reason, the MSP is typically selected by customers who are competing in markets where standardized products can be successful. If customers are competing in markets that value differentiation and feature innovation, the SSP package may be preferable, assuming the customer is able to dedicate the necessary engineering resources.

The MSP layer consists of the components listed below.

- **MW/Codecs**: Flow controllers for video encoding/decoding/transcoding, still picture capturing/decoding, audio recording/decoding, external track (e.g., GPS information) recording/decoding, etc. This also includes arbitration mechanisms (pipelines) coordinating codes working together for multi-stream in/out and synchronization (e.g., A/V sync).
- **MW/FIFO**: Bitstream information dispatcher, which handles multiple bitstream client features.
- **MW/Data**: Data flows including muxer/demuxer/editor, cached file read/write scheduler, network transfer controller, DCF indexing (file naming) system, etc. This also includes arbitration mechanisms (pipelines) coordinating multiple muxer/demuxer instances running simultaneously.
- **MW Unit Test**: Unit tests for all middleware modules.

In a typical MSP release, customers will receive the following:

1. **SSP Libraries**
2. **MSP Libraries**
3. **Unit Test Source Code**
4. **ARD/APP Source Code** (Including its Applibs)
5. **Common Service Source Code**: Frequently-used small utilities
6. **Image Quality Utility Libraries**: Utilities for calibration, AE/AWB/ADJ scheduling, bitrate monitoring

Note that DCF (file naming rules) and GUI drawing utilities are included in the ARD/APP's Applib.

2.1.2.3 Ambarella Reference Design (ARD)

The Ambarella Reference Design (ARD) system refers to the logically separated text section where the top-level functionality of the system exists. It is the main entrance point to the system and provides generic features for reference.

The ARD design is based on market segments, such as **Connected** (Section 2.1.2.3.1). These ARDs/APPs share the same low-level protocols, including SSP, MSP and drivers; however, they diversify in terms of their visible feature sets.

Of the available A12 SDK packages, the ARD system typically offers the shortest time to market. Because modularized and generic flows are both provided, only minimal engineering resources are required (e.g., to modify GUI or GUI flows) when using the ARD system, assuming the selected ARD matches the customers' product requirements exactly.

It should be noted that **the feature set of a given ARD is fixed.**

The ARD/APP layer consists of the components listed below.

- **ARD/APP Applib**: Reference code showing how to construct a specific function by using middleware APIs, such as the materials to configure a video recorder and how to control it.
  - Some middleware modules only provide frameworks, while the ARD/APP Applib includes implementations (e.g., DCF file naming rule, graphics engine).
  - By default, ARD/APP Applib is released as a library.
- **ARD/APP Source Code**: Control/UI flows for generic functions.
  - While the ARD/APP Applib provides modules to configure and control middleware components, control/UI flows can be used to diversify products in terms of user experience.

In a typical ARD release, customers will receive the following:

1. **SSP Libraries**
2. **MSP Libraries**
3. **ARD/APP Source Code**
4. **ARD/APP's Applib Libraries**
5. **Common Service Source Code**: Frequently-used small utilities
6. **Image Quality Utility Libraries**: Utilities for calibration, AE/AWB/ADJ scheduling, bitrate monitoring

2.1.2.3.1 Connected APP

Connected APP is designed to demonstrate generic functions of the Ambarella SDK, such as video encoding, still capture and playback. Connected APP provides a production-wise framework and the simplest flow required to construct a feature. Because Connected APP is not market-specific, all productions can begin from this starting point, making it easier for customers to differentiate their products in a crowded marketplace.

2.2 SDK 6.2.002: Middleware Support Package

Specifications for the A12 Middleware Support Package are shown below. Note that new or updated features for this release have been starred and marked in green.

- Video IN/OUT Support List
  - Sensor
    - Sony IMX206
    - OmniVision OV4689
    - Aptina AR0230
    - Aptina AR0330 Parallel
    - Panasonic MN34120
    - TI TVP5150 (YUV Input)
  - Display
    - 1080p60 HDMI
    - LCD
- Function List
  - Microcode (ucode) Bonding Check
  - Standby Mode
  - Fast File Download
  - Video Encode
    - Stream Matrix Table
      - OmniVision OV4689

NTSC	Encode Resolution	Second Stream Support	
		1920x1080p30	1280x720p30
Main Stream Support	★ 2560x1440p30 HDR	√	√
	2560x1440p30	√	√
	2560x1440p60	No Second Stream	
	★ 1920x1080p30 HDR	√	√
	1920x1080p30	√	√
	1920x1080p60	√	√
	1280x720p30		√
	1280x720p60		√

Table 2-1. OmniVision OV4689 Encode Performance - NTSC.

PAL	Encode Resolution	Second Stream Support	
		1920x1080p25	1280x720p25
Main Stream Support	★ 2560x1440p25 HDR	√	√
	2560x1440p25	√	√
	2560x1440p50		√
	★ 1920x1080p25 HDR	√	√
	1920x1080p25	√	√
	1920x1080p50	√	√
	1280x720p25		√
	1280x720p50		√

Table 2-2. OmniVision OV4689 Encode Performance - PAL.

- Video Encode (Continued)
  - Stream Matrix Table
    - Sony IMX206

NTSC	Encode Resolution	Second Stream Support	
		1920x1080p30	1280x720p30
Main Stream Support	1920x1080p30	√	√
	1920x1080p60	√	√
	1280x720p30		√
	1280x720p60		√
	1280x720p120		√

Table 2-3. Sony IMX206 Encode Performance - NTSC.

PAL	Encode Resolution	Second Stream Support	
		1920x1080p25	1280x720p25
Main Stream Support	1920x1080p25	√	√
	1920x1080p50	√	√
	1280x720p25		√
	1280x720p50		√
	1280x720p100		√

Table 2-4. Sony IMX206 Encode Performance - PAL.

- ◦ Video Encode (Continued)
  - Stream Matrix Table
    - Aptina AR0230

NTSC	Encode Resolution	Second Stream Support	
		1920x1080p30	1280x720p30
Main Stream Support	1920x1080p30	√	√
	1920x1080p60	√	√
	1280x720p30		√
	1280x720p60		√

Table 2-5. Aptina AR0230 Encode Performance - NTSC.

PAL	Encode Resolution	Second Stream Support	
		1920x1080p25	1280x720p25
Main Stream Support	1920x1080p25	√	√
	1920x1080p50	√	√
	1280x720p25		√
	1280x720p50		√

Table 2-6. Aptina AR0230 Encode Performance - PAL.

- ◦ Video Encode (Continued)
  - Stream Matrix Table
    - Aptina AR0330 Parallel

NTSC	Encode Resolution	Second Stream Support	
		1920x1080p30	1280x720p30
Main Stream Support	1920x1080p30	√	√
	1280x720p30		√
	1280x720p60		√

Table 2-7. Aptina AR0330 Parallel Encode Performance - NTSC.

PAL	Encode Resolution	Second Stream Support	
		1920x1080p25	1280x720p25
Main Stream Support	1920x1080p25	√	√
	1280x720p25		√
	1280x720p50		√

Table 2-8. Aptina AR0330 Parallel Encode Performance - PAL.

- ◦ Video Encode (Continued)
  - Stream Matrix Table
    - Panasonic MN34120

NTSC	Encode Resolution	Second Stream Support	
		1920x1080p30	1280x720p30
Main Stream Support	2560x1440p30	√	√
	1920x1080p30	√	√
	1920x1080p60	√	√
	1280x720p30		√
	1280x720p60		√
	1280x720p120		√

Table 2-9. Panasonic MN34120 Encode Performance - NTSC.

- ◦ Video Encode (Continued)
  - Stream Matrix Table
    - TI TVP5150

NTSC	Encode Resolution	Second Stream Support
Main Stream Support	720x480i30	No Second Stream

Table 2-10. TI TVP5150 Encode Performance - NTSC.

PAL	Encode Resolution	Second Stream Support
Main Stream Support	720x576i25	No Second Stream

Table 2-11. TI TVP5150 Encode Performance - PAL.

- Video Encode (Continued)
  - ★ High Dynamic Range
  - ★ Temporal Adjustment
  - PIV
    - Interpolation up to 3840x2160
  - Video Thumbnail
  - Bitrate Control
    - Constant Bitrate
    - Variable Bitrate Monitor
  - Date/Time Stamp
  - QP/AQP Control
  - Digital Zoom 4X
  - Time-Lapsed Encoding
  - 180-degree Rotation
  - Slow Shutter
    - 0.5x
  - Dynamic OB
- Image Tuning
  - Video Tuning
    - Raw to H.264
  - iTuner Script Tuning
  - Amage Tuning
  - IQ Tuning
- Still Capture
  - RAW Capture
  - RAW to YUV
    - LISO
    - MISO
    - HISO
    - Width Interpolation up to 7680
  - RAW to RAW
    - 3A Statistics
  - YUV to JPEG
    - Bitrate Control
  - Burst Capture
  - PES Capture
  - AEB Capture
  - 180-degree Rotation
  - Dynamic OB
- Video Decode
  - Forward Normal/Fast/Slow

  - Backward Normal/Fast/Slow
  - Step
  - Time Search
  - PB Zoom
  - Pause/Resume
- Still Decode
  - Scrolling
  - Cropping
  - Blending
- Calibration
  - Black Level Correction
  - Bad Pixel Correction
  - Chroma Abbreviation
  - Warp
  - Vignette
  - White Balance (with Flash Calibration)
  - Audio
  - Gyro
- Audio
  - Encode : AAC/PCM
  - Decode : AAC/PCM
- OSD
  - 8 bit
  - 16 bit (Performance Bound)
  - 32 bit (Performance Bound)
- Format
  - EXIF
  - MP4 Mux/Demux/Recovery
  - MOV Mux/Demux/Recovery
- DCF
  - Index/Sorting Infrastructure
- Cached File System
  - System for File Search/Status Request
  - Async File Read/Write
- AmbaLink/NetFIFO
  - Streaming via Wi-Fi
  - Net Control via Wi-Fi
  - Notifier Between Two Operating Systems
  - Linux NetFIFO Status Report
  - Information for Online Playback Request
  - Information for Bandwidth Adjustment

- Boot Mode
    - NAND Boot
    - eMMC Boot
    - SPI-NOR Boot
- File System
    - FAT16
    - FAT32
    - ExFAT

2.3 SDK 6.2.002: Connected App

Specifications for the A12 Connected App are shown below. Note that new or updated features for this release have been starred and marked in green.

- Video IN/OUT Support List
  - Sensor
    - OmniVision OV4689
    - Sony IMX206
  - Display
    - Encode:
      - LCD
    - Decode:
      - LCD + 1080p60 HDMI
      - LCD + 480i Composite
- Function List
  - Video Encode
    - Main Stream: (Saved to the SD card)

Sensor: OmniVision OV4689	Encode Performance
Mode: Normal	2560x1440 50fps 16:9 2560x1440 30fps 16:9 1920x1080 60fps 16:9 1920x1080 30fps 16:9 1280x720 60fps 16:9
★ Mode: HDR	2560x1440 30fps 16:9 1920x1080 30fps 16:9

Table 2-12. OmniVision OV4689 Encode Performance for the Main Stream - Normal Mode.

Sensor: Sony IMX206	Encode Performance
Mode: Normal	1920x1080 60fps 16:9 1920x1080 30fps 16:9 1280x720 30fps 16:9
Mode: High Frame Rate	1280x720 120fps 16:9

Table 2-13. Sony IMX206 Encode Performance for the Main Stream - Normal and High Frame Rate Modes.

    - Secondary Stream (For Streaming and Saving to the SD Card)
        - 1280x720p30/p25 4 Mbps
    - Variable Bitrate Control : 0.75x~1.25x
    - PIV (Size is the current capture size of sensor mode; no stamp)
        - ★ HDR mode is not supported
    - Slow Shutter
        - 0.5x (Frame rate > 30)
    - Date/Time Stamp
    - Loop Encoding
    - Event Recording
- ◦ Still Capture
    - Single Capture
    - Burst Capture
    - PES Capture
    - Date/Time Stamp
- ◦ Video Decode
    - Forward Normal/Fast/Slow
    - Backward Normal/Fast/Slow
    - Step
    - Pause/Resume
- ◦ Still Decode
    - Single Photo Playback
    - Thumbnail Playback
- ◦ Calibration
    - Black Level Correction
    - Bad Pixel Correction
    - Chroma Abbreviation
    - Warp
    - Vignette
    - White Balance
- ◦ Audio
    - Encode: AAC
    - Decode: AAC
- ◦ OSD
    - 8-bit OSD
- ◦ Format
    - JPEG and EXIF
    - MP4 Mux/Demux/Recovery
- ◦ DCF
    - Date / Time Naming Rule
- ◦ USB

        - Mass Storage
    - Control
        - Button
        - IR Remote Control
        - Ambalink Network Control
    - AmbaLink
        - Liveview Streaming
        - Upload/Download File
        - Video Playback Streaming
        - Seamless Playback Streaming
        - Get IDR Frame from Clip
        - Get Thumbnail / Full Image of Photo
        - Amba Remote Camera Debug Functions
    - Miscellaneous
        - GPS Information
        - Partial Load

2.4 SDK 6.2.002: SDK 6.2.002: Known Issues

- (Section 2.4.1) Known Issues: Middleware Support Package
- (Section 2.4.2) Known Issues: Connected App
- (Section 2.4.3) Known Issues: Image Quality (IQ)
- (Section 2.4.4) Known Issues: Image Kernel
- (Section 2.4.5) Known Issues: AmbaUSB
- (Section 2.4.6) Known Issues: Amage

2.4.1 Known Issues: Middleware Support Package

Bug ID	Description
28313	[MSP]: Bitrate Average is out-of-setting when executing test BRC settings with some resolutions.
29914	[MSP]: The photos are brighter than LCD preview when executing snapshot.
30608	[MSP]: Second file bitrate is out-of-setting when performing Change Average Bitrate during encoding with dual.
30739	[MSP]: TV outputs no audio when performing AV encode then playing with HDMI output.
30743	[MSP]: Sound is too low when decoding on EVK or PC with AV encode.
30781	[MSP]: TV preview displayed color shift when decoding photo by CVBS.
30800	[MSP]: Preview has broken area on upper left when decoding the thumbnail photo.
30816	[MSP]: The letterbox has color image when decoding thumbnail photo on EVK/TV.
31026	[MSP]: LCD preview is black when backward decoding the clip with 2560x1920 p30.
31074	[MSP][OB]: Photo has a bad line at right side when performing Single JPEG encode at 3rd times with OB.
31079	[MSP][Hybrid-LISO][OV4689]: Second file is broken when performing time-lapse with dual.
31080	[MSP][Hybrid-LISO]: There is a noise bar at the top of clip when encoding with rotate.

31086	[MSP][Hybrid-LISO]: System stops encoding automatically when performing test bitstream spen with Dual-HD with 1440p30 HDR / 1080p30 HDR.
31138	[MSP][ucode]: FOV incorrect when performing Initial zoom under Hybrid-LISO mode.
31163	[MSP]: Secondary stream is broken on the EVK and PC side when executing secwin 360x200 with dual-stream encoding.
31173	[MSP][Hybrid-LISO]: Random system hangs occur (CPUException Assert) when Video Liveview starts during several single encode test sequences (P: 1/5).
31179	[MSP][Hybrid-LISO]: System hangs (Tera Term prints error message) when performing Initial zoom with dual under Hybrid-LISO mode.
31182	[MSP][Hybrid-LISO]: System hangs (Tera Term prints error message) when performing Initial zoom with dual under Hybrid-LISO mode.
31184	[MSP]][Hybrid-LISO]: Preview shows noise flashing (1sec) when changing sensor mode to HDR 1920x1080 30p / 2560x1440 30p.
31186	[MSP][IMX206]: Main file's bitrate is out-of-setting when performing BRC(VBR) with Dual-HD.
31190	[MSP]: Preview shows pink coloration when performing still encode then typing `t stilldec init` command.
31192	[MSP]: LCD preview shows green flash when Live View starts after standby.
31196	[MSP][Hybrid-LISO][TA]: System hangs (Tera Term prints error message) when performing initial zoom with TA under 1280x720 30p.
31226	[MSP][Hybrid-LISO]: LCD preview displays a bad line when starting Live View at 2560x1440 30p HDR.
31228	[MSP][Hybrid-LISO]: Clip changed color when starting video encode with `thm` function on 1080 30p HDR/1440 30P HDR.
31229	[MSP][Hybrid-LISO]: The stamp is broken on LCD preview when enabling blend.
31212	[MSP][AR0330_PARALLEL]: The screennail.JPEG is broken at bottom when performing single JPEG encode.

Table 2-14. Known Issues: Middleware Support Package.

2.4.2 Known Issues: Connected App

Bug ID	Description
31185	[Connected]: System stops encoding clip automatically (clip does not save to SD card) after encoding for approximately 30 secs to 6 mins with 64-GB SDXC card (Start from 20150625 dailybuild).
31197	[Connected]: System hangs (K_ASSERT at AppLibDCF_GetBrowseMode 1607 / Thumbnail displays no file) when changing to thumbnail mode then powering off/on EVK when SD card has 4 files. (Start from dailybuild_20150625)
31211	[Connected][Seamless_Playback]: VLC player cannot playback next stream when performing seamless playback streaming. (Start from dailybuild_20150604).
31204	[Connected][Taroko][OV4689]: Playback becomes stuck when decoding clips with Video Play All enabled under 32x/64x fast forward/backward. [3/5].
31195	[Connected][OV4689]: Thumbnail displays no file when removing/inserting SD card during camera power off/on.
28775	[Connected]: Stamp information of JPG is unchanged when performing PES capture for 30 seconds.
29909	[Connected]: Playing bar and stamp are overlapping when decoding clip with HDMI or Composite cable.
30381	[Connected]: There is no UI display on the TV output if Composite/HDMI cable is inserted while photo/clip is in a decoding state.
30817	[Connected][OV4689]: The stamp of VF preview has broken area when changing to HDR 2560x1440 p30/ HDR 1920x1080 p30 resolutions.
30893	[Connected][OV4689]: LCD preview has a broken line on the bottom side under HDR1920x1080p30.
31013	[Connected][OV4689]: The video is earlier than the audio on second file under 2560x1440p50 with PIV enabled.
31014	[Connected][OV4689]: Preview shows pink flashing on the first frame when changing resolutions with HDR 1920x1080p30 or HDR 2560x1440 p30.
31019	[Connected][OV4689]: Preview flashes a pinkish scene after camera boot.
31020	[Connected][OV4689]: Preview/Clip flashes a pinkish scene after covering/uncovering the lens.
31021	[Connected][OV4689]: The clip has a broken frame when waving a hand in front of the lens under HDR resolutions.
31050	[Connected]: Sound is too low when decoding on EVK or PC.
31081	[Connected][AR0330_parallel]: Stamp of preview is cut on LCD preview under 1280x720p60.
31183	[Connected][Taroko][OV4689]: System hangs when recording starts before streaming has been enabled.

31214	[Connected]: LCD preview becomes dark when switching resolutions from 720p200 to 1080p60, or from 1080p100 to 720p200.
31221	[Connected][Taroko][OV4689]: System hangs when streaming is not yet enabled to perform movie recovery.

Table 2-15. Known Issues: Connected App.

2.4.3 Known Issues: Image Quality (IQ)

Bug ID	Description
31084	[Connected]: Still EXIF does not match AE information.
31115	[MSP] The quality of some frames is substandard for H.264 files/MP4 files.
31162	[Connected][OV4689]: Still AE is unstable (becomes brighter).
31213	[MSP] There are occasional bad frames on MP4_clip.
31215	[MSP] Brightness of version 6.2.002 is different than 6.2.001 under low light conditions.
31077	[MSP][IMX117] 60p is darker than 30p under low light conditions.

Table 2-16. Known Issues: Image Quality.

2.4.4 Known Issues: Image Kernel

Bug ID	Description
N/A	None

Table 2-17. Known Issues: Image Kernel.

2.4.5 Known Issues: AmbaUSB

Bug ID	Description
N/A	Fail (6) After AmbaUSB information display completes, Tera Term information is delayed by 10~15 seconds.
N/A	Fail (12) AmbaUSB tool shows "Fail to reboot" after loading code for approximately 2 minutes.
N/A	Fail (13) AmbaUSB tool can only connect 1~2 EVK when multiple load code on Windows XP x86.
N/A	Fail (14) Tera Term prints a “Get ptb failed” error message when loading code on Windows OS / Linux x64 / Linux x86.

Table 2-18. Known Issues: AmbaUSB.

2.4.6 Known Issues: Amage

Bug ID	Description
N/A	None

Table 2-19. Known Issues: Amage.

2.5 SDK 6.2.002: QA Test Results

This section provides QA test results for version 6.2.002 of the A12 SDK as follows:

- (Section 2.5.1) Test Parameters: Middleware Support Package
- (Section 2.5.2) Test Results: Middleware Support Package
- (Section 2.5.3) Test Parameters: Connected App
- (Section 2.5.4) Test Results: Connected App
- (Section 2.5.5) Test Parameters: Image Quality (IQ)
- (Section 2.5.6) Test Results: Image Quality (IQ)
- (Section 2.5.7) Test Parameters: Image Kernel
- (Section 2.5.8) Test Results: Image Kernel
- (Section 2.5.9) Test Parameters: AmbaUSB
- (Section 2.5.10) Test Results: AmbaUSB
- (Section 2.5.11) Test Parameters: Amage
- (Section 2.5.12) Test Results: Amage

2.5.1 Test Parameters: Middleware Support Package

Test Parameter	Description
Code Information	Code info: ambalink_sdk_3_10_20150610.xml rtos_20150625_1530.xml u_code_version: Version = 242061 Date = 2015/6/25 API = 241950
Sensor Configuration	rtos_a12_mw_unittest_defconfig (Used IMX206, OV4689, AR0230, MN34120PA, and AR0330 Parallel)
Video Resolution	Express Mode: IMX206 1920x1080p100 OV4689 2560x1440p60 2560x1440p30 AR0230 HDR 1920x1080p30 AR0330 Parallel 1920x1080p25 MN34120PA 1280x720p120 Hybrid Mode: OV4689 2560x1440p30 HDR 1920x1080p25 HDR 1920x1080p50 1920x1080p30 1280x720p50 1280x720p30

Still Capture Size	IMX206 3840x2160 photo OV4689 2688x1512 photo AR0230 1920x1080 photo AR0330 Parallel 2304x1296 photo MN34120PA 4608x3456 photo

Table 2-20. Test Parameters: Middleware Support Package.

2.5.2 Test Results: Middleware Support Package

Feature		Test Result	Note
Display	1080p60 HDMI	Pass	Bug 30739
	LCD	Pass	Bug 31184
Standby Mode	Minimal PLL Clocks / Wake up by INT	Pass	Bug 31192
Video Encode	YUV Input/Output	Pass	
	PIV – Interpolation 3840x2160 , Video Thumbnail	Pass	
	Bitrate Control - Constant Bitrate , Variable Bitrate Monitor	Pass	Bug 28313 Bug 30608 Bug 31186
	Date/Time Stamp	Pass	
	QP/AQP Control	Pass	
	Digital Zoom 4X	Pass	Bug 31179 Bug 31182 Bug 31196
	Time-Lapse Encode	Pass	Bug 31079
	180' Rotation	Pass	Bug 31080
	Slow Shutter – 0.5x	Pass	
	Dynamic OB	Pass	
	High Dynamic Range	Pass	Bug 31138
	Temporal Adjustment	Pass	

Still Capture	RAW Capture	Pass	Bug 29914
	RAW to YUV – LISO/HISO	Pass	
	RAW to YUV – Interpolation Max Width 7680	Pass	
	RAW to RAW – 3A Statistics	Pass	
	YUV to JPEG – Bitrate Control	Pass	
	Burst Capture	Pass	
	PES Capture	Pass	
	AEB Capture	Pass	
	180' Rotation	Pass	
	Dynamic OB	Pass	Bug 31074
Video Decode	Forward/Backward Normal/Fast/Slow , Step , Time Search , PB Zoom , Pause/Resume	Pass	Bug 31026
Still Decode	Scrolling , Cropping , Blending	Pass	Bug 30800 Bug 30816
Calibration	Black Level Correction , Bad Pixel Correction , Vignette , White Balance (with Flash Calibration)	Pass	
Display	Color Space Conversion	Pass	
Audio	Encode/Decode : AAC/PCM	Pass	Bug 30743
OSD	8-bit , 16-bit (Performance Bound) , 32-bit (Performance Bound)	Pass	
Format	EXIF	Pass	
	MP4/MOV MUX/DEMUX/RECOVERY	Pass	
	Trick Record (Fast/Slow Motion)	Pass	
DCF	Standard DCF Rule	Pass	
Cached File System	System for File Search/Status Request	Pass	
	Async File Read/Write	Pass	

AmbaLink / NetFIFO	Streaming via Wi-Fi / Net Control via Wi-Fi / Notifier Between two OS / Linux NetFIFO Status Report	Pass	
Boot mode	NAND Boot , eMMC Boot , SPI-NOR Boot	Pass	
File System	FAT16 , FAT32 , ExFAT	Pass	

Table 2-21. Test Results: Middleware Support Package.

2.5.3 Test Parameters: Connected App

Test Parameter	Description
Code Information	Code info: ambalink_sdk_3_10_20150610.xml rtos_20150625_1530.xml u_code_version: Version = 242061 Date = 2015/6/25 API = 241950
Sensor Configuration	rtos_a12_app_connected_imx206_396MHz_defconfig rtos_a12_app_connected_ov4689_taroko_defconfig rtos_a12_app_connected_mn34120_defconfig rtos_a12_app_connected_ar0230_taroko_defconfig rtos_a12_app_connected_ar0330_parallel_taroko_defconfig rtos_a12_app_connected_imx117_taroko_defconfig

Table 2-22. Test Parameters: Connected App.

2.5.4 Test Results: Connected App

Feature		Test Result	Note
Display – Decode	LCD (Encode/Decode)	Pass	Bug 31014 Bug 31019 Bug 31020 Bug 31021 Bug 31214
	LCD + 1080p60 HDMI (Decode)	Pass	Bug 29909 Bug 30381
	LCD + 480i Composite	Pass	Bug 29909 Bug 30381

Video Encode	Normal Encode / Loop Encoding	Pass	Bug 30893 Bug 31013
	Second Stream: (For Streaming and Saving to SD Card) 1280x720p30/p25 4 Mbps	Pass	
	Variable Bitrate Control - 0.75x~1.25x	Pass	
	PIV (Size is the current capture size of sensor mode; no stamp. HDR mode is not supported.)	Pass	
	Slow Shutter 0.5x (Frame rate > 30)	Pass	
	Date/Time Stamp	Pass	
	Event Recording	Pass	
Still Capture	Single Capture	Pass	
	Burst Capture	Pass	
	PES Capture	Pass	
	Date/Time Stamp	Pass	Bug 28775 Bug 30817 Bug 31081
Video Decode	Forward – Normal/Fast/Slow , Backward – Normal/Fast/Slow , Step ,Pause/Resume	Pass	Bug 31204
Still Decode	Thumbnail Playback , Single Photo Playback	Pass	
Storage	Format SD Card	Pass	
	Delete File	Pass	
Mode Switch	Video Mode <-> Photo Mode	Pass	
	Encode Mode <-> Decode Mode	Pass	
Calibration	Black Level Correction , Bad Pixel Correction, Chroma Abbreviation , Warp, White Balance, Vignette	Pass	
Audio	Encode/Decode: AAC	Pass	
OSD	8-bit OSD	Pass	

Format	JPEG and EXIF	Pass	
	MP4 Mux/Demux	Pass	
	Recovery	Pass	
DCF	Date Time Naming Rule	Pass	
USB	Mass Storage	Pass	
Control	Button (EVK Dragonfly)	Pass	
	IR Remote Control	Pass	
	Ambalink Network Control	Pass	
AmbaLink (Continued Below)	Liveview Streaming: 1280x720p30 4 Mbps	Pass	
	Playback Streaming	Pass	
	Seamless Playback Streaming	Fail	Bug 31211

AmbaLink (Continued)	AMBA_START_SESSION AMBA_STOP_SESSION AMBA_RESETVF AMBA_STOP_VF AMBA_RECORD_START AMBA_RECORD_STOP AMBA_GET_RECORD_TIME AMBA_SET_SETTING > VIDEO_RESOLUTION AMBA_SET_SETTING > VIDEO_QUALITY AMBA_TAKE_PHOTO AMBA_NOTIFICATION > CONTINUE_BURST_COMPLETE AMBA_CONTINUE_CAPTURE_STOP AMBA_SET_SETTING > CAPTURE_MODE AMBA_DEL_FILE AMBA_LS AMBA_CD AMBA_PWD AMBA_GET_FILE AMBA_CANCEL_FILE_XFER AMBA_NOTIFICATION > GET_FILE_COMPLETE AMBA_PUT_FILE AMBA_NOTIFICATION > PUT_FILE_COMPLETE AMBA_CANCEL_FILE_XFER AMBA_GET_THUMB AMBA_GET_MEDIAINFO AMBA_SET_MEDIA_ATTRIBUTE AMBA_QUERY_SESSION_HOLDER AMBA_FORMAT AMBA_GET_SPACE AMBA_GET_NUMB_FILES AMBA_GET_DEVICEINFO AMBA_SET_SETTING > CAMERA_CLOCK AMBA_GET_SETTING > CAMERA_CLOCK AMBA_SET_SETTING > DEFAULT_SETTING AMBA_WIFI_RESTART AMBA_SET_WIFI_SETTING AMBA_GET_WIFI_SETTING AMBA_WIFI_STOP AMBA_WIFI_START	Pass	
Miscellaneous	GPS Information	Pass	
	Partial Load	Pass	

Table 2-23. Test Results: Connected App.

2.5.5 Test Parameters: Image Quality (IQ)

Test Parameter	Description
uCode Version	[CA9] Version = 242061 [CA9] Date = 2015/6/25 [CA9] API = 241950 [CA9] Silicon = 199 [CA9] init_data = 0x54F980
Reference Firmware	20150525_A12SDK6.2.001_u02_release

Table 2-24. Test Parameters: Image Quality (IQ).

2.5.6 Test Results: Image Quality (IQ)

Sensor	Resolution	Config	MTF50U AVG (LW/PH)	AVG SNR (dB)	Note
AR0230	1080p30	0525_001_MW	2194	31.96	Pass
		0610_002_MW	2300	32.83	
		0625_002_APP	2135	31.42	
		0625_002_MW	2187	32.98	
IMX117	1080p30	0525_001_MW	1051	39.66	Bug 31077 Bug 31215
		0610_002_MW	1064	38.99	
		0625_002_APP	1065	38.78	
		0625_002_MW	1060	39.01	
IMX206	1080p30	0525_001_MW	961	37.50	Bug 31215 Bug 31213
		0610_002_MW	957	37.91	
		0625_002_APP	970	37.08	
		0625_002_MW	949	38.01	
MN34120	1080p30	0525_001_MW	1165	34.20	Pass
		0610_002_MW	1235	34.35	
		0625_002_APP	1216	32.09	
		0625_002_MW	1230	34.26	

OV4689	1080p30	0525_001_MW	1385	37.54	Sharpness worse, noise better
		0610_002_MW	1879	39.67	Update TA, sharp, noise...
		0625_002_APP	1896	33.13	Similar
		0625_002_MW	1874	34.61	Similar
AR0330_P	1080p30	0525_001_MW	1613	N/A	Pass
		0610_002_MW	1667		
		0625_002_APP	1700		
		0625_002_MW	1714		

Table 2-25. Test Results: Image Quality (IQ).

2.5.7 Test Parameters: Image Kernel

Test Parameter	Description

Code Information	Code Info: A12_MWUT_A12SDK_6.2.002_BRANCHDAILY-BUILD_20150625_1530
	
	Version = 242061
	Date = 2015/6/25
	API = 241950
	Silicon = 199
	init_data = 0x54F980

Table 2-26. Test Parameters: Image Kernel.

2.5.8 Test Results: Image Kernel

Feature		Test Result	Note
iTuner (video)	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass	
Amage Tool	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass	
Video Raw Capture	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass	
Video Raw Encode	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass	

Still Raw Capture	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass	
Still Raw Encode	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass	
HDMI Preview	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass	

Table 2-27. Test Results: Image Kernel.

2.5.9 Test Parameters: AmbaUSB

Test Parameter	Description
Code Information	Windows Version: 3.4.6 Linux Version: 3.4.6
Release Date	2015/6/2

Table 2-28. Test Parameters: AmbaUSB.

2.5.10 Test Results: AmbaUSB

Feature	Test Result	Note
AmbaUSB for Windows		
Windows 8.1 x64 : A12	Pass [Fail(6,12,14)]	
Windows 7 x64 : A12	Pass [Fail(6,12,14)]	
Windows 7 x86 : A12	Pass [Fail(6,12,14)]	
Windows XP x86: A12	Pass [Fail(6,12,13,14)]	
AmbaUSB for Linux (Linux version: Ubuntu 14.04)		
Linux x86: A12	Pass [Fail(6,12,14)]	
Linux x64: A12	Pass [Fail(6,12,14)]	

Table 2-29. Test Results: AmbaUSB.

2.5.11 Test Parameters: Amage

Test Parameter	Description

Code Information	Code Info: A12_MWUT_A12SDK_6.2.002_BRANCHDAILY-BUILD_20150625_1530
	
	Version = 242061
	Date = 2015/6/25
	API = 241950
	Silicon = 199
	init_data = 0x54F980

Table 2-30.   Test Parameters: Amage.

2.5.12 Test Results: Amage

Feature		Test Result
MTP Connection	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
Save iTuner	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
Cmp iTuner Image Kernel and Amage	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] Black Level	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] Dynamic Bad Pixel	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning-Calibration] Vignette	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning-Calibration] Static Bad Pixel	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning-Calibration] Warp	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning-Calibration] CA	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] CFA Leakage	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] Tone Curve	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] R2Y	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] Li-Anti-Aliasing	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] WB	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass

[Video Tuning] dgain	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] GbGr	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] LE	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] CC	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] EC	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] MCTF	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] SA	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] ASF	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Video Tuning] Demosaic	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Still Tuning] LISO	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[Still Tuning] HISO	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[IQ Table-AdjToDram] VideoLISO	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[IQ Table-AdjToDram] StillLISO	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[IQ Table-AdjToDram] StillHISO	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[IQ Table-AdjToDram] Photo	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor] Open	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor] Save	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor] Save As	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor] VideoLISO	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor] VideoHISO	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor] StillLISO	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor] StillHISO	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor] Photo	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor] Scene	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass

[AdjBinEditor] Adj Param	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor] Still Param	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor] Image Param	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor] DeStill Param/DeVideo Param	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
[AdjBinEditor]Default	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass
HDMI Preview	IMX206 / OV4689 / MX34120 / AR0230 / AR0330	Pass

Table 2-31. Test Results: Amage.

2.6 SDK 6.2.002: Power Measurement Test Results

2.6.1 Power Measurement Test Parameters: Dragonfly + Sony IMX206

Test Parameter		Description
Hardware	DSP	A12-A0-RH
	BUB	A12 EVK Dragonfly AB120-101-V11A
	DDR	SAMSUNG DDR3_4GX2
	Sensor	IMX206 (P130716008)
	VOUT Type	Single LCD Wintek (P131014022)
	Storage Media	TOSHIBA 32-GB Class 10 UHS-1 SDHC

Environment Parameters	USB	Disconnected
	Object for Record	Worst Case: Complex Scene
	Distance Between Object and Lens	15 cm
	Light Condition:	300 lux 5700K
		Lens Aperture = F2.8
	True RMS Multimeter:	1. Current Measurement: FLUKE 289 (probe line impedance=0.10Ω) 2. Voltage Measurement: FLUKE 289 (probe line impedance=0.10Ω)

Firmware	Code Information	1.\\builder\share\SYSTEM_BRANCH_DAILY_BUILD\20150625\A12SDK_6_2_002_1530\rtos_a12_app_connected_imx206_396MHz_defconfig
		2. uCode:
		Version = 242061
		Date = 2015/6/25
		API = 241950
		Silicon = 199
		init_data = 0x4CC980

Case Configurations	Case 1.1	1920x1080p60 Preview (STOP VF mode) Capture Window: 2304x1296 CFA: 2304x1296 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 40 Clock: Cortex DRAM Core IDSP 504 396 288 224

Case Configurations (Continued)	Case 1.2	1920x1080p60 Dual HD Encode Capture Window: 2304x1296 CFA: 2304x1296 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 42 Clock: Cortex DRAM Core IDSP 504 396 288 132
	Case 2.1	1920x1080p30 Preview (STOP VF mode) Capture Window: 2304x1296 CFA: 2304x1296 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 39 Clock: Cortex DRAM Core IDSP 504 396 288 132
	Case 2.2	1920x1080p30 Dual HD Encode Capture Window: 2304x1296 CFA: 2304x1296 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 40 Clock: Cortex DRAM Core IDSP 504 396 288 132
	Case 3.1	1280x720p120 Preview (STOP VF mode) Capture Window: 1536x864 CFA: 1536x864 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 40 Clock: Cortex DRAM Core IDSP 504 396 288 192

Case Configurations (Continued)	Case 3.2	1280x720p120 Dual HD Encode Capture Window: 1536x864 CFA: 1536x864 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 41 Clock: Cortex DRAM Core IDSP 504 396 288 192
	Case 4.1	1280x720p30 Preview (STOP VF mode) Capture Window: 2304x1296 CFA: 1792x1008 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 37 Clock: Cortex DRAM Core IDSP 504 396 288 116
	Case 4.2	1280x720p30 Dual HD Encode Capture Window: 2304x1296 CFA: 1792x1008 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 38 Clock: Cortex DRAM Core IDSP 504 396 288 116

Table 2-32.   Power Measurement Test Parameters: Dragonfly + Sony IMX206.

2.6.2 Power Measurement Test Results: Dragonfly + Sony IMX206

Case [1]	Power								
	Channel	SOC_ VDD 0.9 V	SOC_ VDDA 1.0 V	SOC_ VDRAM 1.5 V	SOC_ VDDA18 1.8 V	SOC_ VDDA 3.0 V	SOC_ VDD 3.0 V	DRAM_ VDRAM 1.5 V	Total Power (mW)
1.1	Voltage (V)	0.900	1.010	1.507	1.803	3.005	3.006	1.508	888.55
	Current (mA)	453.0	10.3	71.3	13.7	0.3	50.2	123.6	
	Power (mW)	407.8	10.4	107.5	24.7	0.9	150.9	186.4	
1.2	Voltage (V)	0.900	1.010	1.505	1.803	3.005	3.006	1.508	978.83
	Current (mA)	499.0	10.1	77.0	13.9	0.3	51.2	148.2	
	Power (mW)	449.3	10.2	115.9	25.1	0.9	153.9	223.5	
2.1	Voltage (V)	0.900	0.992	1.502	1.803	3.005	3.003	1.502	701.86
	Current (mA)	358.0	10.1	51.9	13.6	0.4	50.2	76.7	
	Power (mW)	322.2	10.0	77.9	24.5	1.2	150.8	115.2	
2.2	Voltage (V)	0.900	0.992	1.500	1.803	3.005	3.003	1.503	771.04
	Current (mA)	390.0	10.2	56.6	13.8	0.3	50.9	97.3	
	Power (mW)	351.2	10.1	84.9	24.9	0.9	152.9	146.2	
3.1	Voltage (V)	0.900	0.991	1.502	1.803	3.004	3.004	1.497	790.81
	Current (mA)	421.0	10.2	56.7	13.6	0.4	50.1	93.7	
	Power (mW)	379.0	10.1	85.1	24.5	1.2	150.5	140.3	

Case [1]	Power								
	Channel	SOC_ VDD 0.9 V	SOC_ VDDA 1.0 V	SOC_ VDRAM 1.5 V	SOC_ VDDA18 1.8 V	SOC_ VDDA 3.0 V	SOC_ VDD 3.0 V	DRAM_ VDRAM 1.5 V	Total Power (mW)
3.2	Voltage (V)	0.900	0.991	1.499	1.803	3.005	3.004	1.496	884.74
	Current (mA)	467.0	10.2	62.8	13.7	0.3	51.3	120.6	
	Power (mW)	420.3	10.1	94.2	24.7	0.9	154.1	180.4	
4.1	Voltage (V)	0.900	1.004	1.503	1.804	3.004	3.005	1.504	610.09
	Current (mA)	320.0	10.1	40.6	14.6	0.4	50.3	48.0	
	Power (mW)	288.1	10.1	61.0	26.3	1.2	151.1	72.2	
4.2	Voltage (V)	0.900	1.004	1.499	1.804	3.004	3.003	1.504	662.46
	Current (mA)	338.0	10.2	44.6	14.6	0.3	50.9	67.2	
	Power (mW)	304.2	10.2	66.8	26.3	0.9	152.9	101.0	

Table 2-33. Power Measurement Test Results: Dragonfly + Sony IMX206.

Notes:

1. Refer to Section 2.6.1 “Power Measurement Test Parameters: Dragonfly + Sony IMX206” for Case definitions.

2.6.3 Power Measurement Test Parameters: Dragonfly + Panasonic MN34120PA

Test Parameter		Description
Hardware	Chip	A12-A0-RH
	BUB	A12 EVK Dragonfly AB120-101-V11A
	DDR	SAMSUNG DDR3_4GX2
	Sensor	MN34120PA (P120313020)
	VOUT Type	Single LCD Wintek (P131014022)
	Storage Media	TOSHIBA 32 GB Class 10 UHS-1 SDHC
Environment Parameters	USB	Disconnected
	Object for Record	Worst Case: Complex Scene
	Distance Between Object and Lens	15 cm
	Light Condition	300 lux 5700K
		Lens Aperture = F2.8
	True RMS Multimeter:	1. Current Measurement: FLUKE 289 (probe line impedance=0.10Ω) 2. Voltage Measurement: FLUKE 289 (probe line impedance=0.10Ω)

Firmware	Code Information	1.\\builder\share\SYSTEM_BRANCH_DAILY_BUILD\20150625\A12SDK_6_2_002_1530\rtos_a12_app_connected_mn34120_defconfig
		2.uCode:
		Version = 242061
		Date = 2015/6/25
		API = 241950
		Silicon = 199
		init_data = 0X4CC980

Case Configurations	Case 1.1	2560x1440p30 Preview (STOP VF mode) Capture Window: 3840x2160 CFA: 2720x1530 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 45 Clock: Cortex DRAM Core IDSP 792 600 396 264
	Case 1.2	2560x1440p30 Dual HD Encode HDR Capture Window: 3840x2160 CFA: 2720x1530 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 47 Clock: Cortex DRAM Core IDSP 792 600 396 264

Table 2-34.   Power Measurement Test Parameters: Dragonfly + Panasonic MN34120PA.

2.6.4 Power Measurement Test Results: Dragonfly + Panasonic MN34120PA

Case [1]	Power								
	Channel	SOC_ VDD 1.0 V	SOC_ VDDA 1.0 V	SOC_ VDRAM 1.5 V	SOC_ VDDA18 1.8 V	SOC_ VDDA 3.0 V	SOC_ VDD 3.0 V	DRAM_ VDRAM 1.5 V	Total Power (mW)
1.1	Voltage (V)	1.000	1.006	1.501	1.799	3.005	3.000	1.501	1037.44
	Current (mA)	603.0	11.5	59.4	14.1	0.3	46.0	112.7	
	Power (mW)	603.2	11.6	89.2	25.4	0.9	138.0	169.2	
1.2	Voltage (V)	1.000	1.006	1.498	1.799	3.005	3.000	1.503	1137.02
	Current (mA)	658.0	11.5	64.0	14.2	0.2	46.7	136.5	
	Power (mW)	658.2	11.6	95.9	25.5	0.6	140.1	205.1	

Table 2-35. Power Measurement Test Results: Dragonfly + Panasonic MN34120PA.

Notes:

1. Refer to Section 2.6.3 "Power Measurement Test Parameters: Dragonfly + Panasonic MN34120PA" for Case definitions.

2.6.5 Power Measurement Test Parameters: Taroko + OmniVision OV4689

Test Parameter		Description
Hardware	Chip	A1505 A12-A1-RH N86RR-AN4
	BUB	A12_EVK_Taroko (P150209108)
	DDR	CHIPSIP NAND (2 Gb) + DDR3L (4 Gb)
	Sensor	OV4689 (P141205029)
	VOUT Type	Single LCD
	Storage Media	ADATA 16 GB SDHC Class 10 UHS-1
Environment Parameters	USB	Disconnected
	Object For Record	Worst Case: Complex Scene
	Distance Between Object and Lens	15 cm
	Light Condition:	300 lux 5700K
		Lens Aperture = F2.8
	True RMS Multimeter:	1. Current Measurement: FLUKE 289 (probe line impedance=0.10Ω) 2. Voltage Measurement: FLUKE 289 (probe line impedance=0.10Ω)

Firmware	Code Information	1. \\builder\share\SYSTEM_BRANCH_DAILY_BUILD\20150625\A12SDK_6_2_002_1530\rtos_a12_app_connected_ov4689_taroko_defconfig
		2. uCode:
		Version = 242061
		Date = 2015/6/25
		API = 241950
		Silicon = 199
		init_data = 0x4CD980

Case Configurations	Case 1.1	2560x1440p50 Preview (STOP VF mode) Capture Window: 2688x1512 CFA: 2688x1512 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 74 Clock: Cortex DRAM Core IDSP 792 600 396 248
	Case 1.2	2560x1440p50 Dual HD Encode Capture Window: 2688x1512 CFA: 2688x1512 Second Stream: 1280x720p25 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 77 Clock: Cortex DRAM Core IDSP 792 600 396 248
	Case 2.1	2560x1440p30 Preview (STOP VF mode) Capture Window: 2688x1512 CFA: 2688x1512 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 63 Clock: Cortex DRAM Core IDSP 792 600 396 240

Case Configurations (Continued)	Case 2.2	2560x1440p30 Dual HD Encode Capture Window: 2688x1512 CFA: 2688x1512 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 72 Clock: Cortex DRAM Core IDSP 792 600 396 240
	Case 3.1	1920x1080p60 Preview (STOP VF mode) Capture Window: 2688x1512 CFA: 2688x1512 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 73 Clock: Cortex DRAM Core IDSP 792 600 396 300
	Case 3.2	1920x1080p60 Dual HD Encode Capture Window: 2688x1512 CFA: 2688x1512 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 77 Clock: Cortex DRAM Core IDSP 792 600 396 300
	Case 4.1	1920x1080p30 Preview (STOP VF mode) Capture Window: 2688x1512 CFA: 2688x1512 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 65 Clock: Cortex DRAM Core IDSP 792 600 396 148.8

Case Configurations (Continued)	Case 4.2	1920x1080p30 Dual HD Encode Capture Window: 2688x1512 CFA: 2688x1512 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 69 Clock: Cortex DRAM Core IDSP 792 600 396 148.8
	Case 5.1	1280x720p60 Preview (STOP VF mode) Capture Window: 2688x1512 CFA: 1792x1008 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 63 Clock: Cortex DRAM Core IDSP 792 600 396 198
	Case 5.2	1280x720p60 Dual HD Encode Capture Window: 2688x1512 CFA: 1792x1008 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 64 Clock: Cortex DRAM Core IDSP 792 600 396 198
	Case 6.1	HDR 2560x1440p30 Preview (STOP VF mode) Capture Window: 2688x3640 CFA: 2688x1512 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 72 Clock: Cortex DRAM Core IDSP 792 600 396 240

Case Configurations (Continued)	Case 6.2	HDR 2560x1440p30 Dual HD Encode Capture Window: 2688x3640 CFA: 2688x1512 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 78 Clock: Cortex DRAM Core IDSP 792 600 396 240
	Case 7.1	HDR 1920x1080p30 Preview (STOP VF mode) Capture Window: 2688x3640 CFA: 2688x1512 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 69 Clock: Cortex DRAM Core IDSP 792 600 396 180
	Case 7.2	HDR 1920x1080p30 Dual HD Encode Capture Window: 2688x3640 CFA: 2688x1512 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 72 Clock: Cortex DRAM Core IDSP 792 600 396 180

Table 2-36. Power Measurement Test Parameters: Taroko + OmniVision OV4689.

2.6.6 Power Measurement Test Results: Taroko + OmniVision OV4689

Case [1]	Power		
	Channel	BATT_ 3.7 V	Total Power (mW)
1.1	Voltage (V)	3.705	2051.03
	Current (mA)	553.6	
1.2	Voltage (V)	3.708	2594.21
	Current (mA)	699.7	
2.1	Voltage (V)	3.703	1731.06
	Current (mA)	467.5	
2.2	Voltage (V)	3.703	2317.34
	Current (mA)	625.8	
3.1	Voltage (V)	3.708	2020.81
	Current (mA)	545.0	
3.2	Voltage (V)	3.708	2425.58
	Current (mA)	654.2	
4.1	Voltage (V)	3.702	1671.26
	Current (mA)	451.4	

Case [1]	Power		
	Channel	BATT_ 3.7 V	Total Power (mW)
4.2	Voltage (V)	3.707	1950.89
	Current (mA)	526.3	
5.1	Voltage (V)	3.703	1662.56
	Current (mA)	449.0	
5.2	Voltage (V)	3.706	1916.48
	Current (mA)	517.1	
6.1	Voltage (V)	3.705	1930.73
	Current (mA)	521.1	
6.2	Voltage (V)	3.699	2540.54
	Current (mA)	686.8	
7.1	Voltage (V)	3.705	1823.60
	Current (mA)	492.2	
7.2	Voltage (V)	3.708	2104.01
	Current (mA)	567.5	

Table 2-37. Power Measurement Test Results: Taroko + OmniVision OV4689.

Notes:

1. Refer to Section 2.6.5 “Power Measurement Test Parameters: Taroko + OmniVision OV4689” for Case definitions.

2.6.7 Power Measurement Test Parameters: Taroko + Aptina AR0230

Test Parameter		Description
Hardware	Chip	A12-A0-RH A1344 NZ4NX-AN1 1N1 A1299
	BUB	A12 EVK Taroko (P150209048)
	DDR	CHIPSIP NAND (2 Gb) + DDR3L (4 Gb)
	Sensor	AR0230 (P150119005)
	VOUT Type	Single LCD
	Storage Media	Lexar Micro SDHC 600x UHS-I Class 10 32-G
Environment Parameters	USB	Disconnected
	Object for Record	Worst Case: Complex Scene
	Distance Between Object and Lens	15 cm
	Light Condition:	300 lux 5700K
	True RMS Multimeter:	1. Current Measurement: FLUKE 289 (probe line impedance=0.10Ω) 2. Voltage Measurement: FLUKE 289 (probe line impedance=0.10Ω)

Firmware	Code Information	1. \\builder\share\SYSTEM_BRANCH_DAILY_BUILD\20150625\A12SDK_6_2_002_1530\rtos_a12_app_connected_ar0230_taroko_defconfig\
		2. uCode:
		Version = 242061
		Date = 2015/6/25
		API = 241950
		Silicon = 199
		init_data = 0x4CC980

Case Configurations	Case 1.1	1920x1080p60 Preview (STOP VF mode) Capture Window: 1920x1080 CFA: 1920x1080 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 70 Temp (T_a): Room Clock: Cortex DRAM Core IDSP 792 600 396 168
	Case 1.2	1920x1080p60 Dual HD Encode HDR Capture Window: 1920x1080 CFA: 1920x1080 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 76 Temp (T_a): Room Clock: Cortex DRAM Core IDSP 792 600 396 168

Table 2-38. Power Measurement Test Parameters: Taroko + Aptina AR0230.

2.6.8 Power Measurement Test Results: Taroko + Aptina AR0230

Case [1]	Power		
	Channel	BATT_ 3.7 V	Total Power (mW)
1.1	Voltage (V)	3.701	2236.33
	Current (mA)	604.3	
1.2 [2]	Voltage (V)	3.694	2600.58
	Current (mA)	704.0	

Table 2-39. Power Measurement Test Results: Taroko + Aptina AR0230.

Notes:

1. Refer to Section 2.6.7 "Power Measurement Test Parameters: Taroko + Aptina AR0230" for Case definitions.

2.6.9 Power Measurement Test Parameters: Taroko + Sony IMX117

Test Parameter		Description
Hardware	Chip	A12-A0-RH A1344 NZ4NX-AN1 1N1 A1299
	BUB	A12 EVK Taroko (P150209048)
	DDR	CHIPSIP NAND (2 Gb) + DDR3L (4 Gb)
	Sensor	IMX117 (P141216014)
	VOUT Type	Single LCD
	Storage Media	Lexar Micro SDHC 600x UHS-I Class 10 32-G
Environment Parameters	USB	Disconnected
	Object For Record	Worst Case: Complex Scene
	Distance Between Object and Lens	15 cm
	Light Condition:	300 lux 5700K
	True RMS Multimeter:	1. Current Measurement: FLUKE 289 (probe line impedance=0.10Ω) 2. Voltage Measurement: FLUKE 289 (probe line impedance=0.10Ω)

Firmware	Code Information	1. \\builder\share\SYSTEM_BRANCH_DAILY_BUILD\20150625\A12SDK_6_2_002_1530\rtos_a12_app_connected_imx117_taroko_defconfig\
		2. uCode:
		Version = 242061
		Date = 2015/6/25
		API = 241950
		Silicon = 199
		init_data = 0x4CC980

Case Configurations	Case 1.1	2560x1440p50 Preview (STOP VF mode) Capture Window: 3840x2160 CFA: 2720x1530 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 74 Temp (T_a): Room Clock: Cortex DRAM Core IDSP 792 600 396 288
	Case 1.2	2560x1440p50 Dual HD Encode Capture Window: 3840x2160 CFA: 2720x1530 Second Stream: 1280x720p25 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 79 Temp (T_a): Room Clock: Cortex DRAM Core IDSP 792 600 396 288

Table 2-40. Power Measurement Test Parameters: Taroko + Sony IMX117.

2.6.10 Power Measurement Test Results: Taroko + Sony IMX117

Case [1]	Power		
	Channel	BATT_3.7 V	Total Power (mW)
1.1	Voltage (V)	3.706	2340.71
	Current (mA)	631.6	
1.2	Voltage (V)	3.704	2882.08
	Current (mA)	778.1	

Table 2-41. Power Measurement Test Results: Taroko + Sony IMX117.

Notes:

1. Refer to Section 2.6.9 “Power Measurement Test Parameters: Taroko + Sony IMX117” for Case definitions.

2.6.11 Power Measurement Test Parameters: Taroko + Aptina AR0330 Parallel

Test Parameter		Description
Hardware	Chip	A12-A0-RH A1344 NZ4NX-AN1 1N1 A1299
	BUB	A12 EVK Taroko (P150209105) - Reworked for parallel sensor
	DDR	CHIPSIP NAND (2 Gb) + DDR3L (4 Gb)
	Sensor	AR0330 (P150123004)
	VOUT Type	Single LCD
	Storage Media	Lexar Micro SDHC 600x UHS-I Class 10 32-G
Environment Parameters	USB	Disconnected
	Object For Record	Worst Case: Complex Scene
	Distance Between Object and Lens	15 cm
	Light Condition:	300 lux 5700K
	True RMS Multimeter:	1. Current Measurement: FLUKE 289 (probe line impedance=0.10Ω) 2. Voltage Measurement: FLUKE 289 (probe line impedance=0.10Ω)

Firmware	Code Information	1. \\builder\share\SYSTEM_BRANCH_DAILY_BUILD\20150625\A12SDK_6_2_002_1530\rtos_a12_app_connected_ar0330_parallel_taroko_defconfig\
		
		2. uCode:
		Version = 242061
		Date = 2015/6/25
		API = 241950
		Silicon = 199
		init_data = 0x4CC980

Case Configurations	Case 1.1	1920x1080p30 Preview (STOP VF mode) Capture Window: 2304x1296 CFA: 2304x1296 Scene: Worst Case - Complex VOUT: LCD On Video Format: N/A GOP Format: N/A Audio Format: N/A Temp (T_c): 63.5 Temp (T_a): Room Clock: Cortex DRAM Core IDSP 792 600 396 132
	Case 1.2	1920x1080p30 Dual HD Encode Capture Window: 2304x1296 CFA: 2304x1296 Second Stream: 1280x720p30 Scene: Worst Case - Complex VOUT: LCD On Video Format: H.264 GOP Format: Info.Video.M = 1 Info.Video.N = 8 Info.Video.GOPSize = 8 Audio Format: AAC Temp (T_c): 65 Temp (T_a): Room Clock: Cortex DRAM Core IDSP 792 600 396 132

Table 2-42.   Power Measurement Test Parameters: Taroko + Aptina AR0330 Parallel.

2.6.12 Power Measurement Test Results: Taroko + Aptina AR0330 Parallel

Case [1]	Power		
	Channel	**BATT_3.7 V**	**Total Power (mW)**
1.1	Voltage (V)	3.701	1784.11
	Current (mA)	482.1	
1.2	Voltage (V)	3.694	1996.98
	Current (mA)	540.6	

Table 2-43. Power Measurement Test Results: Taroko + Aptina AR0330 Parallel.

Notes:

1. Refer to Section 2.6.11 “Power Measurement Test Parameters: Taroko + Aptina AR0330 Parallel” for Case definitions.

Appendix 1 SDK Change List

This appendix provides a summary of the differences between A12 **SDK6.2.001.u02** and **SDK6.2.002** APIs. The appendix is organized as follows:

- (Section A1.1) BossNet
- (Section A1.2) Data Flow

A1.1 BossNet

```
Add (BRCM) bluetooth support.

diff --git a/vendors/ambarella/inc/./bt/amba_bluetooth.h b/vendors/ambarella/inc/./
bt/amba_bluetooth.h
new file mode 100644
index 0000000..89e57c5
--- /dev/null
+++ b/vendors/ambarella/inc/./bt/amba_bluetooth.h
@@ -0,0 +1,238 @@
+/*
+ * Copyright (C) 2014, Ambarella, Inc.
+ *
+ * All rights reserved. No Part of this file may be reproduced, stored
+ * in a retrieval system, or transmitted, in any form, or by any means,
+ * electronic, mechanical, photocopying, recording, or otherwise,
+ * without the prior consent of Ambarella, Inc.
+ */
+
+/*
+ * HAL of Ambarella Bluetooth.
+ */
+
+#ifndef AMBA_BLUETOOTH_H
+#define AMBA_BLUETOOTH_H
+
+#if defined(__cplusplus)
+extern "C" {
+#endif
+
+
+/* Maximum number of service to be added. */
+#define MAX_SERVICE     (8)
+
+#define LEN_DEVICE_NAME (248)
+#define LEN_BDADDR      (6)
+typedef UINT8 bdaddr_t[LEN_BDADDR];
+
+#define LEN_PINCODE     (16)
+
+
+/* Status of Bluetooth Device/Stack */
+typedef enum {
+    AMBA_BT_STATUS_NONE,                /**< BT device/stack is not initialized. */
+    AMBA_BT_STATUS_ENABLED,             /**< BT device/stack is enabled (running).
*/
+    AMBA_BT_STATUS_DISABLED,            /**< BT device/stack is disabled (sleep-
ing). */
+    AMBA_BT_STATUS_FAILED,              /**< BT device/stack do not work well. */
+
+    AMBA_BT_STATUS_CONNECTED,           /**< BT is at connected status. */
+    AMBA_BT_STATUS_DISCONNECTED         /**< BT is at disconnected status. */
+} amba_bt_status_t;
+
+/* BT errno */
+typedef enum {
+    AMBA_BT_EOK,
+    AMBA_BT_ENG,
+    AMBA_BT_EPERM,       /* Operation not permitted */
+    AMBA_BT_EIO,         /* I/O error */
+    AMBA_BT_EAGAIN,      /* Try again */
+    AMBA_BT_EFAULT,      /* Bad address */
+    AMBA_BT_EBUSY,       /* Device or resource busy */
+    AMBA_BT_ENODEV,      /* No such device */
+    AMBA_BT_EINVAL,      /* Invalid argument */
+    AMBA_BT_EFBIG,       /* File too large */
+    AMBA_BT_ETIME,       /* Timer expired */
+    AMBA_BT_ECOMM,       /* Communication error on send */
+    AMBA_BT_ENETDOWN,    /* Network is down */
+    AMBA_BT_ENETUNREACH, /* Network is unreachable */
+    AMBA_BT_ENOTCONN,    /* Transport endpoint is not connected */
+    AMBA_BT_ETIMEDOUT,   /* Connection timed out */
+    AMBA_BT_EUNDEFINED   /* Undefined error */
+} amba_bt_errno_t;
+
+/* Structure to hold any type of UUID. */
+typedef struct
+{
+#define LEN_UUID_NULL (0)
+#define LEN_UUID16    (2)
+#define LEN_UUID32    (4)
+#define LEN_UUID128   (16)
+
+    UINT16 len;      // LEN_UUID_xx
+    union {
+        UINT16 uuid16;
+        UINT32 uuid32;
+        UINT8  uuid128[LEN_UUID128];
+    } uuid;
+} bt_uuid_t;
+
+/* --- GATT --------------------------------------------------------------------*/
+
+#define BT_GATT_PERM_READ             (1 << 0)
```

```
+#define BT_GATT_PERM_READ_ENCRYPTED    (1 << 1)
+#define BT_GATT_PERM_READ_ENC_MITM     (1 << 2)  /* man-in-the-middle (MITM)  pro-
tection */
+#define BT_GATT_PERM_WRITE             (1 << 4)
+#define BT_GATT_PERM_WRITE_ENCRYPTED   (1 << 5)
+#define BT_GATT_PERM_WRITE_ENC_MITM    (1 << 6)
+#define BT_GATT_PERM_WRITE_SIGNED      (1 << 7)
+#define BT_GATT_PERM_WRITE_SIGNED_MITM (1 << 8)
+
+#define BT_GATT_PERM_READ_AUTH         (BT_GATT_PERM_READ_ENCRYPTED)
+#define BT_GATT_PERM_READ_MITM         (BT_GATT_PERM_READ_ENC_MITM)
+#define BT_GATT_PERM_READ_ENCR         (BT_GATT_PERM_READ_ENCRYPTED | BT_GATT_PERM_
READ_ENC_MITM)
+
+#define BT_GATT_PERM_WRITE_AUTH        (BT_GATT_PERM_WRITE_ENCRYPTED | BT_GATT_PERM_
WRITE_SIGNED)
+#define BT_GATT_PERM_WRITE_MITM        (BT_GATT_PERM_WRITE_ENC_MITM | BT_GATT_PERM_
WRITE_SIGNED_MITM)
+#define BT_GATT_PERM_WRITE_ENCR        (BT_GATT_PERM_WRITE_ENCRYPTED | BT_GATT_PERM_
WRITE_ENC_MITM)
+#define BT_GATT_WRITE_SIGNED_PERM      (BT_GATT_PERM_WRITE_SIGNED | BT_GATT_PERM_
WRITE_SIGNED_MITM)
+
+#define BT_GATT_CHAR_PROP_BROADCAST    (1 << 0)
+#define BT_GATT_CHAR_PROP_READ         (1 << 1)
+#define BT_GATT_CHAR_PROP_WRITE_NR     (1 << 2)  /* Write without response */
+#define BT_GATT_CHAR_PROP_WRITE        (1 << 3)
+#define BT_GATT_CHAR_PROP_NOTIFY       (1 << 4)
+#define BT_GATT_CHAR_PROP_INDICATE     (1 << 5)
+#define BT_GATT_CHAR_PROP_AUTH         (1 << 6)
+#define BT_GATT_CHAR_PROP_EXT_PROP     (1 << 7) /* Extended properties. */
+
+typedef struct {
+    bt_uuid_t characteristic;
+    UINT32    char_permission;                  /* BT_GATT_PERM_xxx */
+    UINT32    char_property;                    /* BT_GATT_CHAR_PROP_xxx */
+
+    bt_uuid_t descriptor;
+    UINT32    desc_permission;                  /* BT_GATT_PERM_xxx */
+
+    /* Get after characteristic/descriptor is added. */
+    UINT16    char_attr_id;                     /* attr_id != but ~= handle. */
+    UINT16    desc_attr_id;                     /* attr_id != but ~= handle. */
+} gatt_char_t;
+
+/* Configuration of GATT service */
+typedef struct {
+    bt_uuid_t    primary;            /* Primary service */
+    UINT16       primary_service_id; /* service_id != but ~= handle. */
+
+    UINT8        num_char;           /* Number of characteristic */
+    gatt_char_t *chars;
+
+    /* Callback function by HAL when got GATT Read request. */
+    void (*cbf_gatt_reqRead)(UINT16 attr_id, void *data, UINT16 *len);
+
+    /* Callback function by HAL when got GATT Write request. */
+    void (*cbf_gatt_reqWrite)(UINT16 attr_id, const void *data, UINT16 len);
+} gatt_service_t;
+
+/* --- BT Device management --------------------------------------------------*/
+
+/* For CONFIG_BT_AMBA_PAIRING and CONFIG_BT_AMBA_SMP */
+typedef struct {
+} amba_bt_smp_t;
+
+
+/* Configuration of BT device. */
+typedef struct {
+    /* Name of device. */
+    char  name[LEN_DEVICE_NAME];
+
+    /* BD address: [5]:[4]:[3]:[2]:[1]:[0] */
+    UINT8 bdaddr[LEN_BDADDR];
+
+#define LEN_PIN_CODE (16)
+    UINT8 pincode[LEN_PIN_CODE]; /* Pin Code (upto 128 bits) MSB is 0 */
+    UINT8 pincode_len;           /* Length of PIN code, 4 or 6 */
+
+    /* Max. priority of BT stack tasks when download (initial stage). */
+    UINT8 max_dl_pri;
+
+    /* Max. priority of BT stack tasks after download (working stage). */
+    UINT8 max_wk_pri;
+} amba_bt_conf_t;
+
+
+/* GATT interface */
+typedef struct {
+    /* GATT profile server initialization. */
+    amba_bt_errno_t (*init)(void);
+
+    /* Add service/characteristics/descriptors */
+    amba_bt_errno_t (*addConf)(gatt_service_t *conf);
+
+    /* Get current GATT MTU */
+    UINT16 (*getMtu)(void);
+
+    /* Send Data, return length sent.
+     * if return 0, mean sendData failed.
+     * need_ack: 1 as indication, 0 as notification.
+     * timeout: timeout in ms for retry.
+     */
+    UINT16 (*sendData)(UINT16 attr_id, const void *data, UINT16 len, UINT8 need_
ack,
+                       amba_bt_errno_t *err, UINT32 timeout);
+
+    /*
+     * Callback functions for GATT server
+     */
+    /* Callback function by HAL when GATT connection status changed. */
+    void (*cbf_connection)(amba_bt_status_t conn);
```

```
+} amba_gatt_if_t;
+
+
+/* Ambarella Bluetooth interface. */
+typedef struct {
+    /*
+     * BT device management
+     */
+    amba_bt_conf_t bt_conf;
+
+    /* BT stack initialization. */
+    void (*bt_init)(int log_en);
+
+    /* Get status of BT device/stack. */
+    amba_bt_status_t (*getStatus)(void);
+
+    /* Enable BT service, return AMBA_BT_EOK or AMBA_BT_ENG. */
+    amba_bt_errno_t (*enable)(void);
+
+    /* Disable BT service, return AMBA_BT_EOK or AMBA_BT_ENG. */
+    amba_bt_errno_t (*disable)(void);
+
+    /*
+     * Callback functions for BT device management
+     */
+
+    /* Callback function when get PIN request, need to be filled by user.
+            Max. number of pin code is 16 */
+    void (*cbf_reqPIN)(const UINT8 *bdaddr, UINT8 *len_pin,
+                       UINT8 *pin_code);
+    /*
+     * GATT interface
+     */
+    amba_gatt_if_t *gatt_if;
+
+    /*
+     * BT Stack maintenance
+     */
+    /* enable/disable log of BT stack. */
+    void (*btStack_log)(UINT8 enable);
+} amba_bt_if_t;
+
+
+extern void amba_bt_init(int log_en);
+extern amba_bt_if_t *amba_bt_getIf(void);
+
+#if defined(__cplusplus)
+}
+#endif
+
+#endif /* AMBA_BLUETOOTH_H */
```

1. [purpose] Remove history from file header

```
diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/Json_Utility.h b/vendors/ambarella/inc/./mw/net/Json_Utility.h
index 648c9fc..b316581 100644
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/Json_Utility.h
+++ b/vendors/ambarella/inc/./mw/net/Json_Utility.h
@@ -1,12 +1,7 @@
 /**
- * @file mw/net/netctrl/Json_Utility.h
+ * @file Json_Utility.h
  *
- * Json_Utility header
- *
- * History:
- *    2014/09/03 - [Howie Liu] created file
- *
- * Copyright (C) 2014, Ambarella, Inc.
+ * Copyright (C) 2015, Ambarella, Inc.
  *
  * All rights reserved. No Part of this file may be reproduced, stored
  * in a retrieval system, or transmitted, in any form, or by any means,
diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/NetEventNotifier.h b/vendors/ambarella/inc/./mw/net/NetEventNotifier.h
index 4607af7..adee526 100644
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/NetEventNotifier.h
+++ b/vendors/ambarella/inc/./mw/net/NetEventNotifier.h
@@ -1,10 +1,5 @@
 /**
- * @file inc/mw/net/NetEventNotifier.h
- *
- * NetEventNotifier header
- *
- * History:
- *    2014/10/06 - [Keny Huang] created file
+ * @file NetEventNotifier.h
  *
  * Copyright (C) 2015, Ambarella, Inc.
  *
diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/NetFifo.h b/vendors/ambarella/inc/./mw/net/NetFifo.h
old mode 100755
new mode 100644
index 64aafc6..a5df97b
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/NetFifo.h
+++ b/vendors/ambarella/inc/./mw/net/NetFifo.h
@@ -1,12 +1,7 @@
 /**
- * @file inc/mw/net/NetFifo.h
+ * @file NetFifo.h
  *
```

```
- * NetFifo header
- *
- * History:
- *    2014/08/01 - [Keny Huang] created file
- *
- * Copyright (C) 2014, Ambarella, Inc.
+ * Copyright (C) 2015, Ambarella, Inc.
  *
  * All rights reserved. No Part of this file may be reproduced, stored
  * in a retrieval system, or transmitted, in any form, or by any means,
diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/NetUtility.h b/vendors/ambarella/inc/./mw/net/NetUtility.h
index c6c5a5f..c832aa8 100644
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/NetUtility.h
+++ b/vendors/ambarella/inc/./mw/net/NetUtility.h
@@ -1,12 +1,7 @@
 /**
- * @file inc/mw/net/NetUtility.h
+ * @file NetUtility.h
  *
- * NetFifo header
- *
- * History:
- *    2014/09/29 - [Keny Huang] created file
- *
- * Copyright (C) 2014, Ambarella, Inc.
+ * Copyright (C) 2015, Ambarella, Inc.
  *
  * All rights reserved. No Part of this file may be reproduced, stored
  * in a retrieval system, or transmitted, in any form, or by any means,
```

2. [purpose] Add RTOS BLE support

```
diff --git a/vendors/ambarella/inc/./mw/net/NetBleCfg.h b/vendors/ambarella/inc/./mw/net/NetBleCfg.h
new file mode 100644
index 0000000..c48204c
--- /dev/null
+++ b/vendors/ambarella/inc/./mw/net/NetBleCfg.h
@@ -0,0 +1,90 @@
+/**
+ * @file NetBleCfg.h
+ *
+ * Copyright (C) 2015, Ambarella, Inc.
+ *
+ * All rights reserved. No Part of this file may be reproduced, stored
+ * in a retrieval system, or transmitted, in any form, or by any means,
+ * electronic, mechanical, photocopying, recording, or otherwise,
+ * without the prior consent of Ambarella, Inc.
+ */
+
+#ifndef __AMP_NETBLECFG_H__
```

```
+#define __AMP_NETBLECFG_H__
+
+#include <net/NetCtrl.h>
+#include <../bt/amba_bluetooth.h>
+
+/**
+ * The BLE infomation of NetCtrlCfg
+ */
+typedef struct _AMP_NETBLECFG_BLEINFO_s_ {
+    UINT32 BleWriteCharId;       /**< This chararictic is used to let the phone-App to
write the data to Camera */
+    UINT32 BleReadCharId;        /**< This chararictic is used to let the phone-App to
read the data from Camera */
+    UINT32 MsgBufSize;           /**< The size of message buffer (byte) */
+    amba_bt_if_t *BleInterfaceIns;      /**< the pointer which is indicated the address
of BLE interface instance
+                                        (the instance of the structure “amba_bt_if_t”)*/
+    gatt_service_t *BleGattServiceIns;  /**< the pointer which is indicated the address
of gatt service instance
+                                        (the instance of the structure “gatt_
service_t”)*/
+} AMP_NETBLECFG_BLEINFO_s;
+
+
+/**
+ * @brief  Register this function to the Gatt write service if you want to use the module
“BleNetworkHandler” to
+ *         handle the input when the Ble gets the writing requirement.
+ *         Please Notice that this function just to support the attrId is same as the
char_attr_id or desc_attr_id
+ *         of chararictic which id is same as BleWriteCharId .
+ * @return NULL
+ */
+extern void AmpNetBleCfg_GattWriteHandler(UINT16 attrId, const void *data, UINT16 len);
+
+/**
+ * @brief  This function is used to get the  Ble information in the module “BleNetwork-
Handler”.
+ * @param [in] The pointer is used to indicate the memory which is saved the returned ble
information instance
+ *             (the instance of the structure “AMP_NETBLECFG_BLEINFO_s”)
+ *
+ * @return 0 - successfully
+ * @return negative value - AMP_NETCTRL_ERROR_e
+ */
+extern int AmpNetBleCfg_GetBleInfo(AMP_NETBLECFG_BLEINFO_s *bleInfoIns);
+
+/**
+ * @brief  This function is used to set the Ble informaton to the module
+ *         “Ble_Network_Handler” after initialized.
+ *         The error code will be returned if the module “BleNetworkHandler”
+ *         has been initialized so that you should call the release fx. in the module
“NetCtrl”
+ *         , set the BLE info, and then call the init fx. in the module “NetCtrl” again.
+ *
+ * @param [in] The pointer is used to indicate the address of the ble information in-
stance
+ *             (the instance of the structure “AMP_NETBLECFG_BLE_INFO_s”)
+ *              Notice that it just let one set of Ble information
+ *              existed at the same time.
+ *
+ * @return 0 - all Ble informaion has been set successfully
+ * @return negative value - AMP_NETCTRL_ERROR_e
+ */
+extern int AmpNetBleCfg_SetBleInfo(AMP_NETBLECFG_BLEINFO_s *bleInfoIns);
+
+/**
+ * @brief  This function is used to reset the resource but not to clean “BLE information”
+ *          in the module “Ble_Network_Handler”.
+ *          Notice that the message buffer just is formatted but not deleted
+ *          and the task “Ble_Msg_Daemon” still keep alive
+ *
+ * @return 0 - OK
+ * @return negative value - AMP_NETCTRL_ERROR_e
+ */
+extern int AmpNetBleCfg_Reset(void);
+
+/**
+ * @brief  This function is used to clean the message buffer in the module
+ *          “Ble_Network_Handler”.
+ *          Notice that the message buffer just is formatted but not deleted.
+ *
+ * @return 0 - OK
+ * @return negative value - AMP_NETCTRL_ERROR_e
+ */
+//extern int AmpNetBleCfg_ResetMsgBuf(void);
+
+
+#endif
diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/NetCtrl.h b/vendors/ambarella/inc/./mw/net/NetCtrl.h
old mode 100755
new mode 100644
index 3cb15df..534cade
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/NetCtrl.h
+++ b/vendors/ambarella/inc/./mw/net/NetCtrl.h
@@ -1,12 +1,7 @@
 /**
- * @file inc/mw/net/NetCtrl.h
+ * @file NetCtrl.h
  *
- * NetCtrl header
- *
- * History:
- *    2014/08/15 - [Howie Liu] created file
- *    2015/01/15 - [Janet Liu] re-design the software architecture and re-define APIs
- * Copyright (C) 2014, Ambarella, Inc.
+ * Copyright (C) 2015, Ambarella, Inc.
  *
  * All rights reserved. No Part of this file may be reproduced, stored
  * in a retrieval system, or transmitted, in any form, or by any means,
@@ -247,6 +242,28 @@ extern int AmpNetCtrl_Init(AMP_NETCTRL_INIT_CFG_s *initCfg);
  */
 extern int AmpNetCtrl_Release(void);
```

```
+/**
+ * Initialize the server for Linux commands such as LS, CD, PWD, and so on.
+ * Please call this function first if you want to call those functions which
+ * will use the shell command in the linux such as AmpNetCtrl_LnxLs and so on.
+ *
+ * @param [in] NULL
+ * @return 0 - OK, others - AMP_NETCTRL_ERROR_e
+ *
+ */
+extern int AmpNetCtrl_LnxCmdInit(void);
+
+/**
+ * Release the server and resources of Linux commands such as LS, CD, PWD, and so on.
+ * Please call this function if you don’t want to call those functions which
+ * will use the shell command in the linux anymore.
+ *
+ * @param [in] NULL
+ * @return 0 - OK, others - AMP_NETCTRL_ERROR_e
+ *
+ */
+extern int AmpNetCtrl_LnxCmdRelease(void);
+

 /**
  * Send LS command for NetCtrl to list the contents of the directory,
@@ -517,7 +534,7 @@ extern int AmpNetCtrl_DataSvc_RegRecvCb(AMP_NETCTRL_DATASVC_HDLR_INFO_s *hdlrInf
  * @return 0 - OK, others - AMP_NETCTRL_ERROR_e
  */

-extern int AmpNetCtrl_DataSvc_Send(AMP_NETCTRL_DATASVC_HDLR_INFO_s *hdlrInfo, AMP_NETCTRL_DATASVC_DATA_s *dataReq,
+extern int AmpNetCtrl_DataSvc_Send(AMP_NETCTRL_DATASVC_HDLR_INFO_s *hdlrInfo, AMP_NETCTRL_DATASVC_DATA_s *dataReq,
                                    AMP_NETCTRL_DATASVC_RESULT_s *dataResult);
 /**
  * Get the current status of data server
@@ -529,7 +546,7 @@ extern int AmpNetCtrl_DataSvc_Send(AMP_NETCTRL_DATASVC_HDLR_INFO_s *hdlrInfo, AM
  * @return 0 - OK, others - AMP_NETCTRL_ERROR_e
  */

-extern int AmpNetCtrl_DataSvc_GetStatus(AMP_NETCTRL_DATASVC_HDLR_INFO_s *hdlrInfo,
+extern int AmpNetCtrl_DataSvc_GetStatus(AMP_NETCTRL_DATASVC_HDLR_INFO_s *hdlrInfo,
                                         AMP_NETCTRL_DATASVC_DEST_INFO_s *destInfo,
                                         int *retResult);

@@ -546,7 +563,7 @@ extern int AmpNetCtrl_DataSvc_GetStatus(AMP_NETCTRL_DATASVC_HDLR_INFO_s *hdlrInf
  */

 extern int AmpNetCtrl_DataSvc_CancelDataTrans(AMP_NETCTRL_DATASVC_HDLR_INFO_s *hdlrInfo,
-                                              AMP_NETCTRL_DATASVC_CANCEL_TRANS_s *cancelTrans,
+                                              AMP_NETCTRL_DATASVC_CANCEL_TRANS_s *cancelTrans,
                                               AMP_NETCTRL_DATASVC_CANCEL_RESULT_s *can-
```

```
celResult);

 /**
@@ -559,7 +576,7 @@ extern int AmpNetCtrl_DataSvc_CancelDataTrans(AMP_NETCTRL_DATASVC_
HDLR_INFO_s *h
  * @return 0 - OK, others - AMP_NETCTRL_ERROR_e
  */

-extern int AmpNetCtrl_DataSvc_CloseConnection(AMP_NETCTRL_DATASVC_HDLR_INFO_s *hdlrInfo,
+extern int AmpNetCtrl_DataSvc_CloseConnection(AMP_NETCTRL_DATASVC_HDLR_INFO_s *hdlrInfo,
                                               AMP_NETCTRL_DATASVC_DEST_INFO_s *destInfo,
                                               int *retResult);
```

```
3. [purpose] Refine EOL of RPC header from MSDOS-type to UNIX-type

diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/rpcprog/AmbaIPC_RpcProg_LU_CmdHndlr.h b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_LU_CmdHndlr.h
old mode 100644
new mode 100755
index 3c661b1..89a31e7
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/
mw/net/rpcprog/AmbaIPC_RpcProg_LU_CmdHndlr.h
+++ b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_LU_CmdHndlr.h
@@ -1,66 +1,63 @@
-/**
- * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_Lu_CmdHndlr.h
- *
- * Header file for NetCtrl RPC Services
- *
- * History:
- *    2014/09/02 - [Howie Liu] created file
- *
- * Copyright (C) 2014, Ambarella, Inc.
- *
- * All rights reserved. No Part of this file may be reproduced, stored
- * in a retrieval system, or transmitted, in any form, or by any means,
- * electronic, mechanical, photocopying, recording, or otherwise,
- * without the prior consent of Ambarella, Inc.
- */
-
-#ifndef _RPC_PROG_LU_CMD_HNDLR_H_
-#define _RPC_PROG_LU_CMD_HNDLR_H_
-
-#include “AmbaIPC_Rpc_Def.h”
-
-#define LU_CMD_HNDLR_PROG_ID       0x20000006
-#define     LU_CMD_HNDLR_VER              1
-
-/* Procedure ID */
-#define LU_CMD_HNDLR_PROC          1
-
```

```
-/* Supported Linux service */
-#define LU_CMD_LS                  0
-#define LU_CMD_CD                  1
-#define LU_CMD_PWD                 2
-#define LU_CMD_GET_WIFI_STAT       3
-#define LU_CMD_GET_WIFI_CFG        4
-#define LU_CMD_SET_WIFI_CFG        5
-#define LU_CMD_START_WIFI          6
-#define LU_CMD_STOP_WIFI           7
-#define LU_CMD_RESTART_WIFI        8
-
-/**
- * linux command operation
- */
-typedef struct _LU_LNXCMDHNDLR_CMD_s_ {
-    unsigned int LuCmd;         /**< linux command id */
-    unsigned int CmdSize;       /**< the string length of the command */
-    unsigned int OutSize;       /**< the maximum size of the output */
-    void* Result;                    /**< the result of execute linux command */
-    char Cmd[1024];               /**< command string */
-} LU_LNXCMDHNDLR_CMD_s;
-
-/**
- * result of linux command operation
- */
-typedef struct _LU_LNXCMDHNDLR_DATA_s_ {
-    int Rval;                               /**< result of execute linux command */
-    unsigned int ResultSize;       /**< size of the result */
-} LU_LNXCMDHNDLR_DATA_s;
-
-/**
- * [in] LU_LNXCMDHNDLR_CMD_s
- * [out] LU_LNXCMDHNDLR_DATA_s
- */
-int LU_LnxCmdHndlr_Svc(LU_LNXCMDHNDLR_CMD_s *pArg, AMBA_IPC_SVC_RESULT_s *pResult);
-AMBA_IPC_REPLY_STATUS_e LU_LnxCmdHndlr_Clnt(LU_LNXCMDHNDLR_CMD_s *pArg, LU_LNXCMDHNDLR_
DATA_s *pResult, int Clnt);
-
-#endif /* RPC_PROG_LU_CMD_HNDLR_H */
-
+/**
+ * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_Lu_CmdHndlr.h
+ *
+ * Header file for NetCtrl RPC Services
+ *
+ * Copyright (C) 2014, Ambarella, Inc.
+ *
+ * All rights reserved. No Part of this file may be reproduced, stored
+ * in a retrieval system, or transmitted, in any form, or by any means,
+ * electronic, mechanical, photocopying, recording, or otherwise,
+ * without the prior consent of Ambarella, Inc.
+ */
+
+#ifndef _RPC_PROG_LU_CMD_HNDLR_H_
+#define _RPC_PROG_LU_CMD_HNDLR_H_
+
+#include “AmbaIPC_Rpc_Def.h”
```

```
+
+#define LU_CMD_HNDLR_PROG_ID       0x20000006
+#define    LU_CMD_HNDLR_VER              1
+
+/* Procedure ID */
+#define LU_CMD_HNDLR_PROC          1
+
+/* Supported Linux service */
+#define LU_CMD_LS                  0
+#define LU_CMD_CD                  1
+#define LU_CMD_PWD                 2
+#define LU_CMD_GET_WIFI_STAT       3
+#define LU_CMD_GET_WIFI_CFG        4
+#define LU_CMD_SET_WIFI_CFG        5
+#define LU_CMD_START_WIFI          6
+#define LU_CMD_STOP_WIFI           7
+#define LU_CMD_RESTART_WIFI        8
+
+/**
+ * linux command operation
+ */
+typedef struct _LU_LNXCMDHNDLR_CMD_s_ {
+    unsigned int LuCmd;         /**< linux command id */
+    unsigned int CmdSize;       /**< the string length of the command */
+    unsigned int OutSize;       /**< the maximum size of the output */
+    void* Result;                    /**< the result of execute linux command */
+    char Cmd[1024];               /**< command string */
+} LU_LNXCMDHNDLR_CMD_s;
+
+/**
+ * result of linux command operation
+ */
+typedef struct _LU_LNXCMDHNDLR_DATA_s_ {
+    int Rval;                               /**< result of execute linux command */
+    unsigned int ResultSize;       /**< size of the result */
+} LU_LNXCMDHNDLR_DATA_s;
+
+/**
+ * [in] LU_LNXCMDHNDLR_CMD_s
+ * [out] LU_LNXCMDHNDLR_DATA_s
+ */
+int LU_LnxCmdHndlr_Svc(LU_LNXCMDHNDLR_CMD_s *pArg, AMBA_IPC_SVC_RESULT_s *pResult);
+AMBA_IPC_REPLY_STATUS_e LU_LnxCmdHndlr_Clnt(LU_LNXCMDHNDLR_CMD_s *pArg, LU_LNXCMDHNDLR_DATA_s *pResult, int Clnt);
+
+#endif /* RPC_PROG_LU_CMD_HNDLR_H */
+
diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/rpcprog/AmbaIPC_RpcProg_LU_DataReq.h b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_LU_DataReq.h
index e6e0022..544c27b 100755
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/rpcprog/AmbaIPC_RpcProg_LU_DataReq.h
+++ b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_LU_DataReq.h
@@ -1,125 +1,123 @@
-/**
- * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_Lu_DataReq.h
- *
- * Header file for NetCtrl RPC Services
- *
- * History:
- *    2014/09/02 - [Howie Liu] created file
- *
- * Copyright (C) 2014, Ambarella, Inc.
- *
- * All rights reserved. No Part of this file may be reproduced, stored
- * in a retrieval system, or transmitted, in any form, or by any means,
- * electronic, mechanical, photocopying, recording, or otherwise,
- * without the prior consent of Ambarella, Inc.
- */
-
-#ifndef _RPC_PROG_LU_DATA_REQ_H_
-#define _RPC_PROG_LU_DATA_REQ_H_
-
-#include “AmbaIPC_Rpc_Def.h”
-
-#define LU_DATA_REQ_PROG_ID                0x20000007
-#define LU_DATA_REQ_VER                         1
-
-/**
- * struct of data request info
- */
-typedef struct _LU_DATASVC_DATA_s_ {
-    unsigned int MsgId;                         /**< send data type*/
-    char Filepath[512];                         /**< filepath */
-    char Md5sum[32];                          /**< md5sum */
-    unsigned long long Offset;                /**< file offset */
-    unsigned long long FetchFilesize;     /**< fetch file size */
-    unsigned int BlockSize;                  /**< the size of block (Kbyte)*/
-    unsigned int MaximalSpeed;   /**< the maximal speed of transmitting file (Kbyte/sec)*/
-    char *BufferAddr;                          /**< data buffer */
-    unsigned char ClientInfo[128];         /**< client identifier */
-    char TransportType[16];                 /**< transport protocol type */
-} LU_DATASVC_DATA_s;
-
-/**
- * struct of data request result
- */
-typedef struct _LU_DATASVC_RESULT_s_ {
-    int Rval;                                           /**< result value */
-    unsigned long long RemSize;             /**< remain file size */
-    unsigned long long TotalFilesize;       /**< total file size */
-} LU_DATASVC_RESULT_s;
-
-/* Procedure ID */
-#define LU_DATA_REQ_PROC                       1
-
-/**
- * [in] LU_DATASVC_DATA_s
- * [out] LU_DATASVC_RESULT_s
- */
-int LU_DataReq_Svc(LU_DATASVC_DATA_s *pArg, AMBA_IPC_SVC_RESULT_s *pResult);
```

```
-AMBA_IPC_REPLY_STATUS_e LU_DataReq_Clnt(LU_DATASVC_DATA_s *pArg, LU_DATASVC_RESULT_s
*pResult, int Clnt);
-
-/* Procedure ID */
-#define LU_DATA_GET_STATUS_PROC            2
-
-/**
- * struct of data transmission status
- */
-typedef struct _LU_DATA_GETSTATUS_s_ {
-    unsigned char ClientInfo[128];        /**< client identifier */
-    char TransportType[16];                /**< transport protocol type */
-} LU_DATA_GETSTATUS_s;
-
-/**
- * [in] LU_DATA_GETSTATUS_s
- * [out] int
- */
-int LU_Data_GetStatus_Svc(LU_DATA_GETSTATUS_s *pArg, AMBA_IPC_SVC_RESULT_s *pResult);
-AMBA_IPC_REPLY_STATUS_e LU_Data_GetStatus_Clnt(LU_DATA_GETSTATUS_s *pArg, int *pResult,
int Clnt);
-
-/* Procedure ID */
-#define LU_DATA_CANCEL_TRANS_PROC          3
-
-/**
- * cancel data transmission info
- */
-typedef struct _LU_DATA_CANCEL_TRANS_s_ {
-    int MsgId;                                   /**< data request type */
-    unsigned char ClientInfo[128];        /**< client identifier */
-    char TransportType[16];                /**< transport protocol type */
-    unsigned long long SentSize;        /**< size of file (AMBA_PUT_FILE) data portion */
-} LU_DATA_CANCEL_TRANS_s;
-
-/**
- * cancel data transmission result
- */
-typedef struct _LU_DATA_CANCEL_RESULT_s_ {
-    int Rval;                                        /**< result value */
-    unsigned long long TransSize;        /**< transferred size */
-} LU_DATA_CANCEL_RESULT_s;
-
-/**
- * [in] LU_DATA_CANCEL_TRANS_s
- * [out] LU_DATA_CANCEL_RESULT_s
- */
-int LU_Data_CancelTrans_Svc(LU_DATA_CANCEL_TRANS_s *pArg, AMBA_IPC_SVC_RESULT_s *pRe-
sult);
-AMBA_IPC_REPLY_STATUS_e LU_Data_CancelTrans_Clnt(LU_DATA_CANCEL_TRANS_s *pArg, LU_DATA_
CANCEL_RESULT_s *pResult, int Clnt);
-
-/* Procedure ID */
-#define LU_DATA_CLOSE_CONNECTION_PROC            4
-
-/**
- * struct of close data connection
- */
-typedef struct _LU_DATA_CLOSE_CONNECT_s_ {
-    unsigned char ClientInfo[128];        /**< client identifier */
-    char TransportType[16];                /**< transport protocol type */
-} LU_DATA_CLOSE_CONNECT_s;
-
-/**
- * [in] LU_DATA_CLOSE_CONNECT_s
- * [out] int
- */
-int LU_Data_CloseConnection_Svc(LU_DATA_CLOSE_CONNECT_s *pArg, AMBA_IPC_SVC_RESULT_s
*pResult);
-AMBA_IPC_REPLY_STATUS_e LU_Data_CloseConnection_Clnt(LU_DATA_CLOSE_CONNECT_s *pArg, int
*pResult, int Clnt);
-
-#endif /* RPC_PROG_LU_DATA_REQ_H */
-
+/**
+ * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_Lu_DataReq.h
+ *
+ * Header file for NetCtrl RPC Services
+ *
+ *
+ * Copyright (C) 2014, Ambarella, Inc.
+ *
+ * All rights reserved. No Part of this file may be reproduced, stored
+ * in a retrieval system, or transmitted, in any form, or by any means,
+ * electronic, mechanical, photocopying, recording, or otherwise,
+ * without the prior consent of Ambarella, Inc.
+ */
+
+#ifndef _RPC_PROG_LU_DATA_REQ_H_
+#define _RPC_PROG_LU_DATA_REQ_H_
+
+#include “AmbaIPC_Rpc_Def.h”
+
+#define LU_DATA_REQ_PROG_ID                0x20000007
+#define LU_DATA_REQ_VER                         1
+
+/**
+ * struct of data request info
+ */
+typedef struct _LU_DATASVC_DATA_s_ {
+    unsigned int MsgId;                          /**< send data type*/
+    char Filepath[512];                          /**< filepath */
+    char Md5sum[32];                           /**< md5sum */
+    unsigned long long Offset;                /**< file offset */
+    unsigned long long FetchFilesize;    /**< fetch file size */
+    unsigned int BlockSize;                  /**< the size of block (Kbyte)*/
+    unsigned int MaximalSpeed;   /**< the maximal speed of transmitting file (Kbyte/sec)*/
+    char *BufferAddr;                           /**< data buffer */
+    unsigned char ClientInfo[128];        /**< client identifier */
+    char TransportType[16];                /**< transport protocol type */
+} LU_DATASVC_DATA_s;
+
+/**
+ * struct of data request result
+ */
```

```
+typedef struct _LU_DATASVC_RESULT_s_ {
+    int Rval;                                           /**< result value */
+    unsigned long long RemSize;            /**< remain file size */
+    unsigned long long TotalFilesize;      /**< total file size */
+} LU_DATASVC_RESULT_s;
+
+/* Procedure ID */
+#define LU_DATA_REQ_PROC                       1
+
+/**
+ * [in] LU_DATASVC_DATA_s
+ * [out] LU_DATASVC_RESULT_s
+ */
+int LU_DataReq_Svc(LU_DATASVC_DATA_s *pArg, AMBA_IPC_SVC_RESULT_s *pResult);
+AMBA_IPC_REPLY_STATUS_e LU_DataReq_Clnt(LU_DATASVC_DATA_s *pArg, LU_DATASVC_RESULT_s
*pResult, int Clnt);
+
+/* Procedure ID */
+#define LU_DATA_GET_STATUS_PROC            2
+
+/**
+ * struct of data transmission status
+ */
+typedef struct _LU_DATA_GETSTATUS_s_ {
+    unsigned char ClientInfo[128];        /**< client identifier */
+    char TransportType[16];                /**< transport protocol type */
+} LU_DATA_GETSTATUS_s;
+
+/**
+ * [in] LU_DATA_GETSTATUS_s
+ * [out] int
+ */
+int LU_Data_GetStatus_Svc(LU_DATA_GETSTATUS_s *pArg, AMBA_IPC_SVC_RESULT_s *pResult);
+AMBA_IPC_REPLY_STATUS_e LU_Data_GetStatus_Clnt(LU_DATA_GETSTATUS_s *pArg, int *pResult,
int Clnt);
+
+/* Procedure ID */
+#define LU_DATA_CANCEL_TRANS_PROC          3
+
+/**
+ * cancel data transmission info
+ */
+typedef struct _LU_DATA_CANCEL_TRANS_s_ {
+    int MsgId;                                   /**< data request type */
+    unsigned char ClientInfo[128];        /**< client identifier */
+    char TransportType[16];                /**< transport protocol type */
+    unsigned long long SentSize;        /**< size of file (AMBA_PUT_FILE) data portion */
+} LU_DATA_CANCEL_TRANS_s;
+
+/**
+ * cancel data transmission result
+ */
+typedef struct _LU_DATA_CANCEL_RESULT_s_ {
+    int Rval;                                        /**< result value */
+    unsigned long long TransSize;        /**< transferred size */
+} LU_DATA_CANCEL_RESULT_s;
+
+/**
```

```
+ * [in] LU_DATA_CANCEL_TRANS_s
+ * [out] LU_DATA_CANCEL_RESULT_s
+ */
+int LU_Data_CancelTrans_Svc(LU_DATA_CANCEL_TRANS_s *pArg, AMBA_IPC_SVC_RESULT_s *pRe-
sult);
+AMBA_IPC_REPLY_STATUS_e LU_Data_CancelTrans_Clnt(LU_DATA_CANCEL_TRANS_s *pArg, LU_DATA_
CANCEL_RESULT_s *pResult, int Clnt);
+
+/* Procedure ID */
+#define LU_DATA_CLOSE_CONNECTION_PROC             4
+
+/**
+ * struct of close data connection
+ */
+typedef struct _LU_DATA_CLOSE_CONNECT_s_ {
+    unsigned char ClientInfo[128];        /**< client identifier */
+    char TransportType[16];                /**< transport protocol type */
+} LU_DATA_CLOSE_CONNECT_s;
+
+/**
+ * [in] LU_DATA_CLOSE_CONNECT_s
+ * [out] int
+ */
+int LU_Data_CloseConnection_Svc(LU_DATA_CLOSE_CONNECT_s *pArg, AMBA_IPC_SVC_RESULT_s
*pResult);
+AMBA_IPC_REPLY_STATUS_e LU_Data_CloseConnection_Clnt(LU_DATA_CLOSE_CONNECT_s *pArg, int
*pResult, int Clnt);
+
+#endif /* RPC_PROG_LU_DATA_REQ_H */
+
diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarel-
la/inc/mw/net/rpcprog/AmbaIPC_RpcProg_LU_EventNotifier.h b/vendors/ambarella/inc/./mw/net/
rpcprog/AmbaIPC_RpcProg_LU_EventNotifier.h
old mode 100644
new mode 100755
index fd7916c..da419e2
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/
mw/net/rpcprog/AmbaIPC_RpcProg_LU_EventNotifier.h
+++ b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_LU_EventNotifier.h
@@ -1,51 +1,48 @@
-/**
- * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_LU_EventNotifier.h
- *
- * Header file for EventNotifier RPC Services (Linux side)
- *
- * History:
- *    2014/10/03 - [Keny Huang] created file
- *
- * Copyright (C) 2014, Ambarella, Inc.
- *
- * All rights reserved. No Part of this file may be reproduced, stored
- * in a retrieval system, or transmitted, in any form, or by any means,
- * electronic, mechanical, photocopying, recording, or otherwise,
- * without the prior consent of Ambarella, Inc.
- */
-#ifndef _RPC_PROG_LU_EVENTNOTIFIER_H_
-#define _RPC_PROG_LU_EVENTNOTIFIER_H_
```

```
-
-#include “AmbaIPC_Rpc_Def.h”
-
-//RPC_INFO definition
-#define LU_EVENTNOTIFIER_PROG_ID    0x20000009
-#define LU_EVENTNOTIFIER_HOST AMBA_IPC_HOST_LINUX
-#define LU_EVENTNOTIFIER_VER (1)
-#define LU_EVENTNOTIFIER_DEFULT_TIMEOUT (0) //AMBA_IPC_ASYNCHRONOUS
-#define LU_EVENTNOTIFIER_NAME “AmbaEventNotifier_LUSVC”
-
-typedef struct _LU_EVENTNOTIFIER_MSGBLK_s_ {
-    unsigned int msg;
-    unsigned int param_len;
-    unsigned char param[512];
-} LU_EVENTNOTIFIER_MSGBLK_s;
-
-//============ RPC_FUNC definition ============
-enum _LU_EVENTNOTIFIER_FUNC_e_ {
-     LU_EVENTNOTIFIER_FUNC_NOTIFY = 1,
-
-     LU_EVENTNOTIFIER_FUNC_AMOUNT
-};
-
-/**
- * [[AMBA_IPC_ASYNCHRONOUS]]
- * [in] LU_EVENTNOTIFIER_MSGBLK_s
- * [out] int
- */
-AMBA_IPC_REPLY_STATUS_e LU_EVENTNOTIFIER_Notify_Clnt(LU_EVENTNOTIFIER_MSGBLK_s *pArg, int
*pResult, int Clnt );
-void LU_EVENTNOTIFIER_Notify_Svc(LU_EVENTNOTIFIER_MSGBLK_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
-
-
-#endif /* _RPC_PROG_LU_EVENTNOTIFIER_H_ */
-
+/**
+ * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_LU_EventNotifier.h
+ *
+ * Header file for EventNotifier RPC Services (Linux side)
+ *
+ * Copyright (C) 2014, Ambarella, Inc.
+ *
+ * All rights reserved. No Part of this file may be reproduced, stored
+ * in a retrieval system, or transmitted, in any form, or by any means,
+ * electronic, mechanical, photocopying, recording, or otherwise,
+ * without the prior consent of Ambarella, Inc.
+ */
+#ifndef _RPC_PROG_LU_EVENTNOTIFIER_H_
+#define _RPC_PROG_LU_EVENTNOTIFIER_H_
+
+#include “AmbaIPC_Rpc_Def.h”
+
+//RPC_INFO definition
+#define LU_EVENTNOTIFIER_PROG_ID    0x20000009
+#define LU_EVENTNOTIFIER_HOST AMBA_IPC_HOST_LINUX
```

```
+#define LU_EVENTNOTIFIER_VER (1)
+#define LU_EVENTNOTIFIER_DEFULT_TIMEOUT (0) //AMBA_IPC_ASYNCHRONOUS
+#define LU_EVENTNOTIFIER_NAME “AmbaEventNotifier_LUSVC”
+
+typedef struct _LU_EVENTNOTIFIER_MSGBLK_s_ {
+    unsigned int msg;
+    unsigned int param_len;
+    unsigned char param[512];
+} LU_EVENTNOTIFIER_MSGBLK_s;
+
+//============ RPC_FUNC definition ============
+enum _LU_EVENTNOTIFIER_FUNC_e_ {
+     LU_EVENTNOTIFIER_FUNC_NOTIFY = 1,
+
+     LU_EVENTNOTIFIER_FUNC_AMOUNT
+};
+
+/**
+ * [[AMBA_IPC_ASYNCHRONOUS]]
+ * [in] LU_EVENTNOTIFIER_MSGBLK_s
+ * [out] int
+ */
+AMBA_IPC_REPLY_STATUS_e LU_EVENTNOTIFIER_Notify_Clnt(LU_EVENTNOTIFIER_MSGBLK_s *pArg, int *pResult, int Clnt );
+void LU_EVENTNOTIFIER_Notify_Svc(LU_EVENTNOTIFIER_MSGBLK_s *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
+
+
+#endif /* _RPC_PROG_LU_EVENTNOTIFIER_H_ */
+
diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/rpcprog/AmbaIPC_RpcProg_LU_NetCtrl.h b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_LU_NetCtrl.h
old mode 100644
new mode 100755
index 6b21d3e..79a52b2
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/rpcprog/AmbaIPC_RpcProg_LU_NetCtrl.h
+++ b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_LU_NetCtrl.h
@@ -1,45 +1,42 @@
-/**
- * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_LU_NetCtrl.h
- *
- * Header file for NetCtrl RPC Services
- *
- * History:
- *    2014/09/02 - [Howie Liu] created file
- *
- * Copyright (C) 2014, Ambarella, Inc.
- *
- * All rights reserved. No Part of this file may be reproduced, stored
- * in a retrieval system, or transmitted, in any form, or by any means,
- * electronic, mechanical, photocopying, recording, or otherwise,
- * without the prior consent of Ambarella, Inc.
- */
-
-#ifndef _RPC_PROG_LU_NETCTRL_H_
```

```
-#define _RPC_PROG_LU_NETCTRL_H_
-
-#include “AmbaIPC_Rpc_Def.h”
-
-#define LU_NETCTRL_PROG_ID         0x20000005
-#define    LU_NETCTRL_VER                1
-
-/* Procedure ID */
-#define LU_NETCTRL_REPLY_SVC        1
-
-/**
- * response of network control command
- */
-typedef struct _LU_NETCTRL_DATA_s_ {
-   char *Param;                       /**< json string context */
-   unsigned int ParamSize;      /**< size of json string */
-   unsigned int ClientId;            /**< socket descriptor of accepted client */
-} LU_NETCTRL_DATA_s;
-
-/**
- * [in] LU_NETCTRL_DATA_s
- * [out] int
- */
-int LU_NetCtrl_Reply_Svc(LU_NETCTRL_DATA_s *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
-AMBA_IPC_REPLY_STATUS_e LU_NetCtrl_Reply_Clnt(LU_NETCTRL_DATA_s *pArg, int *pResult, int
nClnt);
-
-#endif /* _RPC_PROG_LU_NETCTRL_H_ */
-
+/**
+ * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_LU_NetCtrl.h
+ *
+ * Header file for NetCtrl RPC Services
+ *
+ * Copyright (C) 2014, Ambarella, Inc.
+ *
+ * All rights reserved. No Part of this file may be reproduced, stored
+ * in a retrieval system, or transmitted, in any form, or by any means,
+ * electronic, mechanical, photocopying, recording, or otherwise,
+ * without the prior consent of Ambarella, Inc.
+ */
+
+#ifndef _RPC_PROG_LU_NETCTRL_H_
+#define _RPC_PROG_LU_NETCTRL_H_
+
+#include “AmbaIPC_Rpc_Def.h”
+
+#define LU_NETCTRL_PROG_ID         0x20000005
+#define    LU_NETCTRL_VER                1
+
+/* Procedure ID */
+#define LU_NETCTRL_REPLY_SVC        1
+
+/**
+ * response of network control command
+ */
+typedef struct _LU_NETCTRL_DATA_s_ {
+   char *Param;                       /**< json string context */
+   unsigned int ParamSize;      /**< size of json string */
+   unsigned int ClientId;            /**< socket descriptor of accepted client */
+} LU_NETCTRL_DATA_s;
+
+/**
+ * [in] LU_NETCTRL_DATA_s
+ * [out] int
+ */
+int LU_NetCtrl_Reply_Svc(LU_NETCTRL_DATA_s *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
+AMBA_IPC_REPLY_STATUS_e LU_NetCtrl_Reply_Clnt(LU_NETCTRL_DATA_s *pArg, int *pResult, int
nClnt);
+
+#endif /* _RPC_PROG_LU_NETCTRL_H_ */
+
diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/
ambarella/inc/mw/net/rpcprog/AmbaIPC_RpcProg_LU_NetFifo.h b/vendors/ambarella/inc/./mw/net/
rpcprog/AmbaIPC_RpcProg_LU_NetFifo.h
old mode 100644
new mode 100755
index 7c484a0..0a15dac
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/
mw/net/rpcprog/AmbaIPC_RpcProg_LU_NetFifo.h
+++ b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_LU_NetFifo.h
@@ -1,86 +1,83 @@
-/**
- * @file inc/mw/net/rpcprog/LU_NetFifo.h
- *
- * Header file for NetFifo RPC Services (RTOS side)
- *
- * History:
- *    2014/08/19 - [Keny Huang] created file
- *
- * Copyright (C) 2014, Ambarella, Inc.
- *
- * All rights reserved. No Part of this file may be reproduced, stored
- * in a retrieval system, or transmitted, in any form, or by any means,
- * electronic, mechanical, photocopying, recording, or otherwise,
- * without the prior consent of Ambarella, Inc.
- */
-#ifndef _RPC_PROG_LU_NETFIFO_H_
-#define _RPC_PROG_LU_NETFIFO_H_
-
-#include “AmbaIPC_Rpc_Def.h”
-
-#ifndef FOREVER
-#define FOREVER 0xFFFFFFFF
-#endif
-
-//RPC_INFO definition
-#define LU_NETFIFO_PROG_ID   0x20000004
-#define LU_NETFIFO_HOST AMBA_IPC_HOST_LINUX
-#define LU_NETFIFO_VER (1)
-#define LU_NETFIFO_DEFULT_TIMEOUT (FOREVER)
-#define LU_NETFIFO_NAME “AMBANETFIFO_LUSVC”
-
-
-typedef enum _LU_NETFIFO_CONTROL_CMD_e_ {
-    LU_NETFIFO_CMD_STARTENC = 1, /**< Start encode. May from STOP_ENC, SWITCHENCSESSION
```

```
or STARTNETPLAY */
-    LU_NETFIFO_CMD_STOPENC, /**< Stop encode then stay in idle. Such as menu operation or
switch to thumbnail mode */
-    LU_NETFIFO_CMD_SWITCHENCSESSION, /**< Stop encode then start another encode session
(VF <-> REC) */
-    LU_NETFIFO_CMD_STARTNETPLAY, /**< Start playback for streaming. set param1 as stream_
id */
-    LU_NETFIFO_CMD_STOPNETPLAY, /**< Stop playback for streaming. set param1 as STARTENC
to indecate APP is recording */
-    LU_NETFIFO_CMD_RELEASE /**< RTOS NetFifo mudule released */
-} LU_NETFIFO_CONTROL_CMD_e;
-
-typedef struct _LU_NETFIFO_CONTROL_ARG_s_ {
-    unsigned int Cmd;
-    unsigned int Param1;
-    unsigned int Param2;
-} LU_NETFIFO_CONTROL_ARG_s;
-
-typedef struct _LU_NETFIFO_FIFOCALLBACK_ARG_s_ {
-    unsigned int FifoHndlr;
-    unsigned int Event;
-    unsigned int InfoPtr;
-} LU_NETFIFO_FIFOCALLBACK_ARG_s;
-
-//============ RPC_FUNC definition ============
-enum _LU_NETFIFO_FUNC_e_ {
-     LU_NETFIFO_FUNC_CONTROLEVENT = 1,
-     LU_NETFIFO_FUNC_FIFOEVENT,
-     LU_NETFIFO_FUNC_FIFOGETWRITEPOINT,
-
-     LU_NETFIFO_FUNC_AMOUNT
-};
-
-/**
- * [in] LU_NETFIFO_CONTROL_ARG_s
- * [out] int
- */
-AMBA_IPC_REPLY_STATUS_e LU_NetFifo_ControlEvent_Clnt(LU_NETFIFO_CONTROL_ARG_s *pArg, int
*pResult, int Clnt );
-void LU_NetFifo_ControlEvent_Svc(LU_NETFIFO_CONTROL_ARG_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
-
-/**
- * [in] LU_NETFIFO_FIFOCALLBACK_ARG_s
- * [out] int
- */
-AMBA_IPC_REPLY_STATUS_e LU_NetFifo_FifoEvent_Clnt(LU_NETFIFO_FIFOCALLBACK_ARG_s *pArg,
int *pResult, int Clnt );
-void LU_NetFifo_FifoEvent_Svc(LU_NETFIFO_FIFOCALLBACK_ARG_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
-
-/**
- * [in] LU_NETFIFO_FIFOCALLBACK_ARG_s
- * [out] int
- */
-AMBA_IPC_REPLY_STATUS_e LU_NetFifo_FifoGetWritePoint_Clnt(LU_NETFIFO_FIFOCALLBACK_ARG_s
*pArg, int *pResult, int Clnt );
-void LU_NetFifo_FifoGetWritePointt_Svc(LU_NETFIFO_FIFOCALLBACK_ARG_s *pArg, AMBA_IPC_SVC_
RESULT_s *pRet);
-
-
-#endif /* _RPC_PROG_LU_NETFIFO_H_ */
-
+/**
+ * @file inc/mw/net/rpcprog/LU_NetFifo.h
+ *
+ * Header file for NetFifo RPC Services (RTOS side)
+ *
+ * Copyright (C) 2014, Ambarella, Inc.
+ *
+ * All rights reserved. No Part of this file may be reproduced, stored
+ * in a retrieval system, or transmitted, in any form, or by any means,
+ * electronic, mechanical, photocopying, recording, or otherwise,
+ * without the prior consent of Ambarella, Inc.
+ */
+#ifndef _RPC_PROG_LU_NETFIFO_H_
+#define _RPC_PROG_LU_NETFIFO_H_
+
+#include “AmbaIPC_Rpc_Def.h”
+
+#ifndef FOREVER
+#define FOREVER 0xFFFFFFFF
+#endif
+
+//RPC_INFO definition
+#define LU_NETFIFO_PROG_ID    0x20000004
+#define LU_NETFIFO_HOST AMBA_IPC_HOST_LINUX
+#define LU_NETFIFO_VER (1)
+#define LU_NETFIFO_DEFULT_TIMEOUT (FOREVER)
+#define LU_NETFIFO_NAME “AMBANETFIFO_LUSVC”
+
+
+typedef enum _LU_NETFIFO_CONTROL_CMD_e_ {
+    LU_NETFIFO_CMD_STARTENC = 1, /**< Start encode. May from STOP_ENC, SWITCHENCSESSION
or STARTNETPLAY */
+    LU_NETFIFO_CMD_STOPENC, /**< Stop encode then stay in idle. Such as menu operation or
switch to thumbnail mode */
+    LU_NETFIFO_CMD_SWITCHENCSESSION, /**< Stop encode then start another encode session
(VF <-> REC) */
+    LU_NETFIFO_CMD_STARTNETPLAY, /**< Start playback for streaming. set param1 as stream_
id */
+    LU_NETFIFO_CMD_STOPNETPLAY, /**< Stop playback for streaming. set param1 as STARTENC
to indecate APP is recording */
+    LU_NETFIFO_CMD_RELEASE /**< RTOS NetFifo mudule released */
+} LU_NETFIFO_CONTROL_CMD_e;
+
+typedef struct _LU_NETFIFO_CONTROL_ARG_s_ {
+    unsigned int Cmd;
+    unsigned int Param1;
+    unsigned int Param2;
+} LU_NETFIFO_CONTROL_ARG_s;
+
+typedef struct _LU_NETFIFO_FIFOCALLBACK_ARG_s_ {
+    unsigned int FifoHndlr;
+    unsigned int Event;
+    unsigned int InfoPtr;
+} LU_NETFIFO_FIFOCALLBACK_ARG_s;
+
+//============ RPC_FUNC definition ============
+enum _LU_NETFIFO_FUNC_e_ {
+     LU_NETFIFO_FUNC_CONTROLEVENT = 1,
+     LU_NETFIFO_FUNC_FIFOEVENT,
+     LU_NETFIFO_FUNC_FIFOGETWRITEPOINT,
+
+     LU_NETFIFO_FUNC_AMOUNT
+};
+
+/**
+ * [in] LU_NETFIFO_CONTROL_ARG_s
+ * [out] int
+ */
+AMBA_IPC_REPLY_STATUS_e LU_NetFifo_ControlEvent_Clnt(LU_NETFIFO_CONTROL_ARG_s *pArg, int
*pResult, int Clnt );
+void LU_NetFifo_ControlEvent_Svc(LU_NETFIFO_CONTROL_ARG_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
+
+/**
+ * [in] LU_NETFIFO_FIFOCALLBACK_ARG_s
+ * [out] int
+ */
+AMBA_IPC_REPLY_STATUS_e LU_NetFifo_FifoEvent_Clnt(LU_NETFIFO_FIFOCALLBACK_ARG_s *pArg,
int *pResult, int Clnt );
+void LU_NetFifo_FifoEvent_Svc(LU_NETFIFO_FIFOCALLBACK_ARG_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
+
+/**
+ * [in] LU_NETFIFO_FIFOCALLBACK_ARG_s
+ * [out] int
+ */
+AMBA_IPC_REPLY_STATUS_e LU_NetFifo_FifoGetWritePoint_Clnt(LU_NETFIFO_FIFOCALLBACK_ARG_s
*pArg, int *pResult, int Clnt );
+void LU_NetFifo_FifoGetWritePointt_Svc(LU_NETFIFO_FIFOCALLBACK_ARG_s *pArg, AMBA_IPC_SVC_
RESULT_s *pRet);
+
+
+#endif /* _RPC_PROG_LU_NETFIFO_H_ */
+
diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarel-
la/inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_DataNotify.h b/vendors/ambarella/inc/./mw/net/
rpcprog/AmbaIPC_RpcProg_RT_DataNotify.h
old mode 100644
new mode 100755
index ca2510f..e06408d
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/
mw/net/rpcprog/AmbaIPC_RpcProg_RT_DataNotify.h
+++ b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_RT_DataNotify.h
@@ -1,48 +1,45 @@
-/**
- * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_DataNotify.h
- *
- * Header file for NetCtrl RPC Services
- *
- * History:
- *    2014/09/02 - [Howie Liu] created file
- *
- * Copyright (C) 2014, Ambarella, Inc.
- *
- * All rights reserved. No Part of this file may be reproduced, stored
- * in a retrieval system, or transmitted, in any form, or by any means,
- * electronic, mechanical, photocopying, recording, or otherwise,
- * without the prior consent of Ambarella, Inc.
- */
-
-#ifndef _RPC_PROG_RT_DATA_NOTIFY_H_
-#define _RPC_PROG_RT_DATA_NOTIFY_H_
-
-#include “AmbaIPC_Rpc_Def.h”
-
-#define RT_DATA_NOTIFY_PROG_ID                  0x10000006
-#define RT_DATA_NOTIFY_VER                      1
-
-/* Procedure ID */
-#define RT_DATA_NOTIFY_PROC                     1
-
-/**
- * data daemon status
- */
-typedef struct _RT_DATASVC_STATUS_s_ {
-    short Status;                           /**< send status of data server */
-    unsigned long long Bytes;         /**< send/recv data bytes */
-    unsigned int Type;                    /**< notification type of data server */
-    unsigned char ClientInfo[128];      /**< client identifier */
-    char TransportType[16];              /**< transport protocol type */
-    unsigned char Md5sum[32];       /**< md5 checksum */
-} RT_DATASVC_STATUS_s;
-
-/**
- * [in] RT_DATASVC_STATUS_s
- * [out] int
- */
-int RT_DataNotify_Svc(RT_DATASVC_STATUS_s *pStatus, AMBA_IPC_SVC_RESULT_s *pRet);
-AMBA_IPC_REPLY_STATUS_e RT_DataNotify_Clnt(RT_DATASVC_STATUS_s *pStatus, int *pResult,
int Clnt);
-
-#endif /* RPC_PROG_DATA_NOTIFY_H */
-
+/**
+ * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_DataNotify.h
+ *
+ * Header file for NetCtrl RPC Services
+ *
+ * Copyright (C) 2014, Ambarella, Inc.
+ *
+ * All rights reserved. No Part of this file may be reproduced, stored
+ * in a retrieval system, or transmitted, in any form, or by any means,
+ * electronic, mechanical, photocopying, recording, or otherwise,
+ * without the prior consent of Ambarella, Inc.
+ */
+
+#ifndef _RPC_PROG_RT_DATA_NOTIFY_H_
+#define _RPC_PROG_RT_DATA_NOTIFY_H_
+
+#include "AmbaIPC_Rpc_Def.h"
+
+#define RT_DATA_NOTIFY_PROG_ID                 0x10000006
+#define RT_DATA_NOTIFY_VER                     1
+
+/* Procedure ID */
+#define RT_DATA_NOTIFY_PROC                    1
+
+/**
+ * data daemon status
+ */
+typedef struct _RT_DATASVC_STATUS_s_ {
+    short Status;                              /**< send status of data server */
+    unsigned long long Bytes;           /**< send/recv data bytes */
+    unsigned int Type;                      /**< notification type of data server */
+    unsigned char ClientInfo[128];        /**< client identifier */
+    char TransportType[16];                /**< transport protocol type */
+    unsigned char Md5sum[32];       /**< md5 checksum */
+} RT_DATASVC_STATUS_s;
+
+/**
+ * [in] RT_DATASVC_STATUS_s
+ * [out] int
+ */
+int RT_DataNotify_Svc(RT_DATASVC_STATUS_s *pStatus, AMBA_IPC_SVC_RESULT_s *pRet);
+AMBA_IPC_REPLY_STATUS_e RT_DataNotify_Clnt(RT_DATASVC_STATUS_s *pStatus, int *pResult,
int Clnt);
+
+#endif /* RPC_PROG_DATA_NOTIFY_H */
+
diff --git a/vendors/ambarella/inc/../../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_EventNotifier.h b/vendors/ambarella/inc/./mw/net/
rpcprog/AmbaIPC_RpcProg_RT_EventNotifier.h
index d5f05c8..ac9e2b0 100755
--- a/vendors/ambarella/inc/../../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/
mw/net/rpcprog/AmbaIPC_RpcProg_RT_EventNotifier.h
+++ b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_RT_EventNotifier.h
@@ -1,53 +1,50 @@
-/**
- * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_EventNotifier.h
- *
- * Header file for EventNotifier RPC Services (RTOS side)
- *
- * History:
- *    2014/08/19 - [Keny Huang] created file
- *
- * Copyright (C) 2014, Ambarella, Inc.
- *
- * All rights reserved. No Part of this file may be reproduced, stored
- * in a retrieval system, or transmitted, in any form, or by any means,
- * electronic, mechanical, photocopying, recording, or otherwise,
- * without the prior consent of Ambarella, Inc.
- */
```

```
-#ifndef _RPC_PROG_RT_EVENTNOTIFIER_H_
-#define _RPC_PROG_RT_EVENTNOTIFIER_H_
-
-#include “AmbaIPC_Rpc_Def.h”
-
-//RPC_INFO definition
-#define RT_EVENTNOTIFIER_PROG_ID    0x10000009
-#define RT_EVENTNOTIFIER_HOST AMBA_IPC_HOST_THREADX
-#define RT_EVENTNOTIFIER_VER (1)
-#define RT_EVENTNOTIFIER_DEFULT_TIMEOUT (0) //AMBA_IPC_ASYNCHRONOUS
-#define RT_EVENTNOTIFIER_NAME “LINK_RPC_SVC_EVENTNOTIFIER”
-
-#define RT_EVENTNOTIFIER_MAX_MSGBLK_PARAM_LEN (512)
-
-typedef struct _RT_EVENTNOTIFIER_MSGBLK_s_ {
-    unsigned int msg;
-    unsigned int param_len;
-    unsigned char param[RT_EVENTNOTIFIER_MAX_MSGBLK_PARAM_LEN];
-} RT_EVENTNOTIFIER_MSGBLK_s;
-
-//============ RPC_FUNC definition ============
-enum _RT_EVENTNOTIFIER_FUNC_e_ {
-    RT_EVENTNOTIFIER_FUNC_NOTIFY = 1,
-
-    RT_EVENTNOTIFIER_FUNC_AMOUNT
-};
-
-
-/**
- * [[AMBA_IPC_ASYNCHRONOUS]]
- * [in] RT_EVENTNOTIFIER_MSGBLK_s
- * [out] int
- */
-AMBA_IPC_REPLY_STATUS_e RT_EVENTNOTIFIER_Notify_Clnt(RT_EVENTNOTIFIER_MSGBLK_s *pArg, int
*pResult, int Clnt );
-void RT_EVENTNOTIFIER_Notify_Svc(RT_EVENTNOTIFIER_MSGBLK_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
-
-
-#endif /* _RPC_PROG_RT_EVENTNOTIFIER_H_ */
+/**
+ * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_EventNotifier.h
+ *
+ * Header file for EventNotifier RPC Services (RTOS side)
+ *
+ * Copyright (C) 2014, Ambarella, Inc.
+ *
+ * All rights reserved. No Part of this file may be reproduced, stored
+ * in a retrieval system, or transmitted, in any form, or by any means,
+ * electronic, mechanical, photocopying, recording, or otherwise,
+ * without the prior consent of Ambarella, Inc.
+ */
+#ifndef _RPC_PROG_RT_EVENTNOTIFIER_H_
+#define _RPC_PROG_RT_EVENTNOTIFIER_H_
+
+#include “AmbaIPC_Rpc_Def.h”
+
+//RPC_INFO definition
+#define RT_EVENTNOTIFIER_PROG_ID    0x10000009
+#define RT_EVENTNOTIFIER_HOST AMBA_IPC_HOST_THREADX
+#define RT_EVENTNOTIFIER_VER (1)
+#define RT_EVENTNOTIFIER_DEFULT_TIMEOUT (0) //AMBA_IPC_ASYNCHRONOUS
+#define RT_EVENTNOTIFIER_NAME “LINK_RPC_SVC_EVENTNOTIFIER”
+
+#define RT_EVENTNOTIFIER_MAX_MSGBLK_PARAM_LEN (512)
+
+typedef struct _RT_EVENTNOTIFIER_MSGBLK_s_ {
+    unsigned int msg;
+    unsigned int param_len;
+    unsigned char param[RT_EVENTNOTIFIER_MAX_MSGBLK_PARAM_LEN];
+} RT_EVENTNOTIFIER_MSGBLK_s;
+
+//============ RPC_FUNC definition ============
+enum _RT_EVENTNOTIFIER_FUNC_e_ {
+    RT_EVENTNOTIFIER_FUNC_NOTIFY = 1,
+
+    RT_EVENTNOTIFIER_FUNC_AMOUNT
+};
+
+
+/**
+ * [[AMBA_IPC_ASYNCHRONOUS]]
+ * [in] RT_EVENTNOTIFIER_MSGBLK_s
+ * [out] int
+ */
+AMBA_IPC_REPLY_STATUS_e RT_EVENTNOTIFIER_Notify_Clnt(RT_EVENTNOTIFIER_MSGBLK_s *pArg, int
*pResult, int Clnt );
+void RT_EVENTNOTIFIER_Notify_Svc(RT_EVENTNOTIFIER_MSGBLK_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
+
+
+#endif /* _RPC_PROG_RT_EVENTNOTIFIER_H_ */
diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/am-
barella/inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_NetCtrl.h b/vendors/ambarella/inc/./mw/net/
rpcprog/AmbaIPC_RpcProg_RT_NetCtrl.h
old mode 100644
new mode 100755
index 4ad2c25..a94140f
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/
mw/net/rpcprog/AmbaIPC_RpcProg_RT_NetCtrl.h
+++ b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_RT_NetCtrl.h
@@ -1,45 +1,42 @@
-/**
- * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_NetCtrl.h
- *
- * Header file for NetCtrl RPC Services
- *
- * History:
- *    2014/09/02 - [Howie Liu] created file
- *
- * Copyright (C) 2014, Ambarella, Inc.
- *
- * All rights reserved. No Part of this file may be reproduced, stored
- * in a retrieval system, or transmitted, in any form, or by any means,
- * electronic, mechanical, photocopying, recording, or otherwise,
- * without the prior consent of Ambarella, Inc.
- */
-
-#ifndef _RPC_PROG_RT_NETCTRL_H_
-#define _RPC_PROG_RT_NETCTRL_H_
-
-#include “AmbaIPC_Rpc_Def.h”
-
-#define RT_NETCTRL_PROG_ID         0x10000005
-#define     RT_NETCTRL_VER                1
-
-/* Procedure ID */
-#define RT_NETCTRL_SEND_SVC       1
-
-/**
- * network control command with json string
- */
-typedef struct _RT_NETCTRL_CMD_s_ {
-   char  Param[1024];    /**< json string of cmd */
-   unsigned int ParamSize;      /**< size of json string to cmd  */
-   unsigned int ClientId;            /**< socket descriptor of accepted client */
-} RT_NETCTRL_CMD_s;
-
-/**
- * [in] RT_NETCTRL_CMD_s
- * [out] int
- */
-int RT_NetCtrl_Send_Svc(RT_NETCTRL_CMD_s *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
-AMBA_IPC_REPLY_STATUS_e RT_NetCtrl_Send_Clnt(RT_NETCTRL_CMD_s *pArg, int *pResult, int
Clnt);
-
-#endif /* _RPC_PROG_RT_NETCTRL_H_ */
-
+/**
+ * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_NetCtrl.h
+ *
+ * Header file for NetCtrl RPC Services
+ *
+ * Copyright (C) 2014, Ambarella, Inc.
+ *
+ * All rights reserved. No Part of this file may be reproduced, stored
+ * in a retrieval system, or transmitted, in any form, or by any means,
+ * electronic, mechanical, photocopying, recording, or otherwise,
+ * without the prior consent of Ambarella, Inc.
+ */
+
+#ifndef _RPC_PROG_RT_NETCTRL_H_
+#define _RPC_PROG_RT_NETCTRL_H_
+
+#include “AmbaIPC_Rpc_Def.h”
+
+#define RT_NETCTRL_PROG_ID         0x10000005
+#define     RT_NETCTRL_VER                1
+
+/* Procedure ID */
+#define RT_NETCTRL_SEND_SVC       1
```

```
+
+/**
+ * network control command with json string
+ */
+typedef struct _RT_NETCTRL_CMD_s_ {
+   char  Param[1024];    /**< json string of cmd */
+   unsigned int ParamSize;      /**< size of json string to cmd  */
+   unsigned int ClientId;            /**< socket descriptor of accepted client */
+} RT_NETCTRL_CMD_s;
+
+/**
+ * [in] RT_NETCTRL_CMD_s
+ * [out] int
+ */
+int RT_NetCtrl_Send_Svc(RT_NETCTRL_CMD_s *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
+AMBA_IPC_REPLY_STATUS_e RT_NetCtrl_Send_Clnt(RT_NETCTRL_CMD_s *pArg, int *pResult, int Clnt);
+
+#endif /* _RPC_PROG_RT_NETCTRL_H_ */
+
diff --git a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_NetFifo.h b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_RT_NetFifo.h
index 1ac2e40..fec725a 100755
--- a/vendors/ambarella/inc/../../../../../A12SDK_6_2_001_u02/rtos/vendors/ambarella/inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_NetFifo.h
+++ b/vendors/ambarella/inc/./mw/net/rpcprog/AmbaIPC_RpcProg_RT_NetFifo.h
@@ -1,353 +1,350 @@
-/**
- * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_NetFifo.h
- *
- * Header file for NetFifo RPC Services (RTOS side)
- *
- * History:
- *    2014/08/19 - [Keny Huang] created file
- *
- * Copyright (C) 2014, Ambarella, Inc.
- *
- * All rights reserved. No Part of this file may be reproduced, stored
- * in a retrieval system, or transmitted, in any form, or by any means,
- * electronic, mechanical, photocopying, recording, or otherwise,
- * without the prior consent of Ambarella, Inc.
- */
-#ifndef _RPC_PROG_RT_NETFIFO_H_
-#define _RPC_PROG_RT_NETFIFO_H_
-
-#include “AmbaIPC_Rpc_Def.h”
-
-#ifndef FOREVER
-#define FOREVER 0xFFFFFFFF
-#endif
-
-//RPC_INFO definition
-#define RT_NETFIFO_PROG_ID   0x10000004
-#define RT_NETFIFO_HOST AMBA_IPC_HOST_THREADX
-#define RT_NETFIFO_VER (1)
-#define RT_NETFIFO_DEFULT_TIMEOUT (FOREVER)
```

```
-#define RT_NETFIFO_NAME “LINK_RPC_SVC_NETFIFO”
-
-//===== mw.h =====
-#define RT_NETFIFO_EVENT_START_NUM (0x04000000) //Should be sync with AMP_FIFO_EVENT_
START_NUM in mw.h
-
-//===== fifo.h =====
-#define RT_NETFIFO_MARK_EOS          0x00FFFFFF
-#define RT_NETFIFO_MARK_EOS_PAUSE    0x00FFFFFE
-
-typedef enum _RT_NETFIFO_EVENT_e_ {
-    RT_NETFIFO_EVENT_DATA_CONSUMED = RT_NETFIFO_EVENT_START_NUM, /**< data consumed event
to data provider (write fifo)*/
-    RT_NETFIFO_EVENT_DATA_EOS, /**< data end of stream to data consumer (read fifo)*/
-    RT_NETFIFO_EVENT_DATA_READY, /**< new data ready to data consumer (read fifo)*/
-    RT_NETFIFO_EVENT_GET_WRITE_POINT, /**< get write pointer info from data consumer
(write fifo)*/
-    RT_NETFIFO_EVENT_RESET_FIFO /**< ask data consumer (write fifo) to reset everything
for bitsfifo reuse*/
-} RT_NETFIFO_EVENT_e;
-
-typedef enum _RT_NETFIFO_FRMAE_TYPE_e_ {
-    RT_NETFIFO_TYPE_MJPEG_FRAME = 0,    ///< MJPEG frame type
-    RT_NETFIFO_TYPE_IDR_FRAME = 1,    ///< Idr frame type
-    RT_NETFIFO_TYPE_I_FRAME = 2,      ///< I frame type
-    RT_NETFIFO_TYPE_P_FRAME = 3,      ///< P frame type
-    RT_NETFIFO_TYPE_B_FRAME = 4,      ///< B frame type
-    RT_NETFIFO_TYPE_JPEG_FRAME = 5,   ///< jpeg main frame
-    RT_NETFIFO_TYPE_THUMBNAIL_FRAME = 6,  ///< jpeg thumbnail frame
-    RT_NETFIFO_TYPE_SCREENNAIL_FRAME = 7, ///< jpeg screennail frame
-    RT_NETFIFO_TYPE_AUDIO_FRAME = 8,      ///< audio frame
-    RT_NETFIFO_TYPE_UNDEFINED = 9,        ///< others
-
-    RT_NETFIFO_TYPE_DECODE_MARK = 101, ///< used when feeding bitstream to dsp. will push
out all frame. */
-    RT_NETFIFO_TYPE_EOS = 255,                  ///< eos bits that feed to raw buffer
-
-    RT_NETFIFO_TYPE_LAST = RT_NETFIFO_TYPE_EOS
-} RT_NETFIFO_FRMAE_TYPE_e;
-
-typedef struct _RT_NETFIFO_s_ {
-    unsigned int nFifoId; /**< the unique of the fifo */
-    void *Ctx; /**< private data of the fifo */
-} RT_NETFIFO_HDLR_s;
-
-typedef struct _RT_NETFIFO_BITS_DESC_s_ {
-    unsigned int SeqNum; /**< sequential number of bits buffer */
-    unsigned long long Pts; /**< time stamp in ticks */
-    unsigned char Type; /**< data type of the entry. see RT_NETFIFO_FRMAE_TYPE_e*/
-    unsigned char Completed; /**< if the buffer content a complete entry */
-    unsigned short  Align; /** data size alignment (in bytes, align = 2^n, n is a integer
)*/
-    unsigned char* StartAddr; /**< start address of data */
-    unsigned int Size; /**< real data size */
-} RT_NETFIFO_BITS_DESC_s;
-
-
-typedef enum _RT_NETFIFO_CFG_INIT_DATA_FETCH_CONDITION_e_{
```

```
-    RT_NETFIFO_CFG_INIT_DISABLE = 0,  /**< no neeed for init data */
-    RT_NETFIFO_CFG_INIT_WITH_TIME, /**< get frames with given length */
-    RT_NETFIFO_CFG_INIT_WITH_NUM_FRAME, /**< get frames with given frame number. ex. if
you set  NumFrame to 100, you will get 100 frames(if there are 100 frames)*/
-} RT_NETFIFO_CFG_INIT_DATA_FETCH_CONDITION_e;
-
-typedef struct _RT_NETFIFO_CFG_INIT_DATA_s_ {
-    unsigned char CreateFifoWithInitData; /**< If create fifo with init data (if data val-
id) */
-
-    RT_NETFIFO_CFG_INIT_DATA_FETCH_CONDITION_e InitCondition; /**< define how to descript
initial data required. */
-
-    union {
-        unsigned long long TimeLength; /**< In normal case, the value of backward fetch
is 0.\n
-         * it means fifo will output frames only after fifo is created.\n
-         * As the valus is not 0, fifo will try to find out the longest valid frames
-         */
-        unsigned long long NumFrame; /**< number of frame */
-    } InitParam; /**< parameter for determinate initial data*/
-
-    RT_NETFIFO_FRMAE_TYPE_e FristFrameType; /**< Frame type of first frame for valid data,
used on video stream to ensure start with Idr. */
-
-    unsigned long long OnCreateFirstFramePts; /** [OUT] Pts of first frame,
0xFFFFFFFFFFFFFFFF if first frame is not valid on create */
-    unsigned long long OnCreateTimeLength; /** [OUT] data length in fifo on create in ms
*/
-} RT_NETFIFO_CFG_INIT_DATA_s;
-
-typedef struct _RT_NETFIFO_CFG_s_ {
-    void *hCodec; /**< the codec the fifo is working on. */
-    unsigned int NumEntries; /**< Number of entries of a fifo */
-    unsigned int IsVirtual; /**< Creating virtual fifo (for data flow) or not (for codec)
*/
-    unsigned int cbEvent; /**< the callback function for fifo event */
-    unsigned int EventDataConsumedThreshold; /**<
-     * the threshold for event RT_NETFIFO_CALLBACK_EVENT_DATA_CONSUMED\n
-     * event only triggered if remain data us under the threshold.
-     * 0 to disable
-     */
-    unsigned int EventDataReadySkipNum; /**<
-     * if not 0, RT_NETFIFO_CALLBACK_EVENT_DATA_READY will not be triggered every frame
encoded.\n
-     * it will be triggered every (eventDataReadySkipNum frame+1) frames
-     */
-    unsigned int cbGetWritePoint; /**<
-     * the callback function invoked when we receive prepareSpace.\n
-     * it should ONLY be used on fifo linked to codec \n
-     * it only take effect on a codec read fifo.\n
-     * the callback should be registered by a decode mgr for all decode codec\n
-     */
-
-    unsigned char SyncRpOnWrite;
-    /**< if 0, rp of base fifo will be updated on remove of virtual fifo.\n
-     *   if 1, rp of base fifo will be updated on write of base fifo.\n
-     *   ONLY take effect on base fifo.
-     */
-
-    unsigned int RawBaseAddr; /**< when sync rp on write is on and RawBaseAddr/RawLimitAddr is given, base fifo will remove desc automatically on overwrite. \n
-                         * only needed on base fifo and sync on write is on
-                         */
-
-    unsigned int RawLimitAddr; /**< when sync rp on write is on and RawBaseAddr/RawLimitAddr is given, base fifo will remove desc automatically on overwrite. \n
-                         * only needed on base fifo and sync on write is on
-                         */
-
-    unsigned long long TickPerSecond; /**< TiackPerSecond used for Pts for the fifo. \n
-    Could be 0 if we whould NOT like to use init data. */
-
-    RT_NETFIFO_CFG_INIT_DATA_s InitData; /**< init status for virtual fifo */
-} RT_NETFIFO_CFG_s;
-
-typedef struct _RT_NETFIFO_INFO_s_ {
-    unsigned int TotalEntries; /**< total number of entries */
-    unsigned int AvailEntries; /**< number of entries with data*/
-} RT_NETFIFO_INFO_s;
-
-//===== NetFifo.h(Muxer.h) =====
-typedef enum _RT_NETFIFO_MEDIA_TRACK_TYPE_e_ {
-    RT_NETFIFO_MEDIA_TRACK_TYPE_VIDEO = 0x01,  /**< The track’s type is Video */
-    RT_NETFIFO_MEDIA_TRACK_TYPE_AUDIO = 0x02,  /**< The track’s type is Audio */
-    RT_NETFIFO_MEDIA_TRACK_TYPE_TEXT = 0x03,   /**< The track’s type is Text */
-    RT_NETFIFO_MEDIA_TRACK_TYPE_MAX = 0x04     /**< Max value, for check use */
-} RT_NETFIFO_MEDIA_TRACK_TYPE_e;
-
-typedef struct _RT_NETFIFO_VIDEO_TRACK_CFG_s_ {
-    unsigned int nGOPSize;            /**< The number of the picture between IDR pictures. */
-    unsigned int nTrickRecDen;        /**< For AmpFormat_ConvertPTS(), the Denominator of the trick record. If is high frame rate, the denominator and numerator are specify the factor than default 30fps, Ex: 120fps -> Den = 1, Num = 4 */
-    unsigned int nTrickRecNum;        /**< For AmpFormat_ConvertPTS(), the Numerator of the trick record. If is high frame rate, the denominator and numerator are specify the factor than default 30fps, Ex: 120fps -> Den = 1, Num = 4 */
-    unsigned int nCodecTimeScale;     /**< TODO: It needs get from H264 bitstream, it is not configurable. */
-    unsigned short  nWidth;              /**< Picture width */
-    unsigned short  nHeight;             /**< Picture height */
-    unsigned short  nM;                  /**< The number of the picture between reference pictures(IDR, I, P) */
-    unsigned short  nN;                  /**< The number of the picture between I pictures */
-    unsigned char bDefault;             /**< The flag defines the track as default video track, if the media hasn’t one video track. */
-    unsigned char nMode;                /**< The value defines the picture mode of the video. It has progressive and interlaced mode. Interlaced mode has Field Per Sample and Frame Per Sample, See AMP_VIDEO_MODE_s */
-    unsigned char bClosedGOP;           /**< The structure of the Close GOP is I P B B P B B. The structure of the Open GOP is I B B P B B, If resume or auto split, the value always is Open GOP. */
```

```
-} RT_NETFIFO_VIDEO_TRACK_CFG_s;
-
-typedef struct _RT_NETFIFO_AUDIO_TRACK_CFG_s_ {
-    unsigned int nSampleRate;          /**< The sample rate(Hz) of the audio track. */
-    unsigned char bDefault;              /**< The flag defines the track as default audio
track, if the media hasn’t one audio track. */
-    unsigned char nChannels;             /**< The number of audio channel. */
-    unsigned char nBitsPerSample;        /**< The per sample size of the audio track. Ex:
8 bits, 16 bits, ....n bits */
-} RT_NETFIFO_AUDIO_TRACK_CFG_s;
-
-typedef struct _RT_NETFIFO_TEXT_TRACK_CFG_s_ {
-    unsigned char bDefault;              /**< The flag defines the track as default text
track, if the media hasn’t one text track. */
-} RT_NETFIFO_TEXT_TRACK_CFG_s;
-
-typedef struct _RT_NETFIFO_MEDIA_TRACK_CFG_s_{
-    unsigned int nMediaId;             /**< The media type of the track. The id is media
id, See AMP_FORMAT_MID_e */
-    unsigned int nTimeScale;           /**< Time scale, the same as the LCM of those of
all tracks */
-    unsigned int nTimePerFrame;        /**< The time of the frame that the unit of it is
Time scale. */
-    unsigned int nInitDelay;           /**< Initial delay time(ms) of the track. */
-    void *hCodec;           /**< the codec which this track is working on. */
-    unsigned char *pBufferBase;            /**< The start address of the FIFO of the
track. User pushes data to the FIFO, the FIFO will write data to the address of the buffer.
*/
-    unsigned char *pBufferLimit;           /**< The end address of the FIFO of the track,
The data can’t write overlap the address, FIFO size = FIFO buffer limit - FIFO buffer
base. */
-    union {
-        RT_NETFIFO_VIDEO_TRACK_CFG_s Video;
-        RT_NETFIFO_AUDIO_TRACK_CFG_s Audio;
-        RT_NETFIFO_TEXT_TRACK_CFG_s Text;
-    } Info;
-    unsigned char nTrackType;            /**< Track type, See AMP_MEDIA_TRACK_TYPE_e */
-} RT_NETFIFO_MEDIA_TRACK_CFG_s;
-
-#define RT_NETFIFO_MAX_TRACK_PER_MEDIA  4   /**< The max track number of a media. */
-
-typedef struct _RT_NETFIFO_MOVIE_INFO_CFG_s_ {
-    RT_NETFIFO_MEDIA_TRACK_CFG_s Track[RT_NETFIFO_MAX_TRACK_PER_MEDIA];
-    unsigned char nTrack;                /**< The number of Tracks in the movie. */
-} RT_NETFIFO_MOVIE_INFO_CFG_s;
-
-//===== self sepcific =====
-typedef struct _RT_NETFIFO_PEEKENTRY_ARG_s_ {
-    RT_NETFIFO_HDLR_s *fifo;
-    unsigned int distanceToLastEntry;
-} RT_NETFIFO_PEEKENTRY_ARG_s;
-
-typedef struct _RT_NETFIFO_REMOVEENTRY_ARG_s_ {
-    RT_NETFIFO_HDLR_s *fifo;
-    unsigned int EntriesToBeRemoved;
-} RT_NETFIFO_REMOVEENTRY_ARG_s;
-
```

```
-typedef struct _RT_NETFIFO_WRITEENTRY_ARG_s_ {
-    RT_NETFIFO_HDLR_s *fifo;
-    RT_NETFIFO_BITS_DESC_s desc;
-} RT_NETFIFO_WRITEENTRY_ARG_s;
-
-typedef struct _RT_NETFIFO_MEDIA_STREAMID_LIST_s_ {
-    int Amount;
-    int StreamID_List[16];
-} RT_NETFIFO_MEDIA_STREAMID_LIST_s;
-
-typedef struct _RT_NETFIFO_PLAYBACK_OP_PARAM_s_ {
-    unsigned int OP;
-    unsigned char Param[128];
-} RT_NETFIFO_PLAYBACK_OP_PARAM_s;
-
-//============ RPC_FUNC definition ============
-enum _RT_NETFIFO_FUNC_e_ {
-    RT_NETFIFO_FUNC_GETDEFAULTCFG = 1,
-    RT_NETFIFO_FUNC_CREATE,
-    RT_NETFIFO_FUNC_DELETE,
-    RT_NETFIFO_FUNC_ERASEALL,
-    RT_NETFIFO_FUNC_GETINFO,
-    RT_NETFIFO_FUNC_PEEKENTRY,
-    RT_NETFIFO_FUNC_REVPEEKENTRY,
-    RT_NETFIFO_FUNC_REMOVEENTRY,
-    RT_NETFIFO_FUNC_PREPAREENTRY,
-    RT_NETFIFO_FUNC_WRITEENTRY,
-    RT_NETFIFO_FUNC_GETMEDIASTREAMID,
-    RT_NETFIFO_FUNC_GETMEDIAINFO,
-    RT_NETFIFO_FUNC_PLAYBACK_OP,
-    RT_NETFIFO_FUNC_REPORTSTATUS,
-
-    RT_NETFIFO_FUNC_AMOUNT
-};
-
-
-/**
- * [in] NULL
- * [out] RT_NETFIFO_CFG_s
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_GetDefaultCfg_Clnt(void *pArg, int *pResult, int Clnt
);
-void RT_NetFifo_GetDefaultCfg_Svc(void *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
-
-/**
- * [in] RT_NETFIFO_CFG_s
- * [out] (RT_NETFIFO_HDLR_s**)
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_Create_Clnt(RT_NETFIFO_CFG_s *pArg, int *pResult, int
Clnt );
-void RT_NetFifo_Create_Svc(RT_NETFIFO_CFG_s *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
-
-/**
- * [in] (RT_NETFIFO_HDLR_s *)
- * [out] int
- */
```

```
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_Delete_Clnt(RT_NETFIFO_HDLR_s **pArg, int *pResult,
int Clnt );
-void RT_NetFifo_Delete_Svc(RT_NETFIFO_HDLR_s **pArg, AMBA_IPC_SVC_RESULT_s *pRet);
-
-/**
- * [in] (RT_NETFIFO_HDLR_s *)
- * [out] int
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_EraseAll_Clnt(RT_NETFIFO_HDLR_s **pArg, int *pResult,
int Clnt );
-void RT_NetFifo_EraseAll_Svc(RT_NETFIFO_HDLR_s **pArg, AMBA_IPC_SVC_RESULT_s *pRet);
-
-/**
- * [in] (RT_NETFIFO_HDLR_s *)
- * [out] RT_NETFIFO_INFO_s
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_GetInfo_Clnt(RT_NETFIFO_HDLR_s **pArg, int *pResult,
int Clnt );
-void RT_NetFifo_GetInfo_Svc(RT_NETFIFO_HDLR_s **pArg, AMBA_IPC_SVC_RESULT_s *pRet);
-
-/**
- * [in] RT_NETFIFO_PEEKENTRY_ARG_s
- * [out] RT_NETFIFO_BITS_DESC_s
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_PeekEntry_Clnt(RT_NETFIFO_PEEKENTRY_ARG_s *pArg, int
*pResult, int Clnt );
-void RT_NetFifo_PeekEntry_Svc(RT_NETFIFO_PEEKENTRY_ARG_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
-
-/**
- * [in] RT_NETFIFO_PEEKENTRY_ARG_s
- * [out] RT_NETFIFO_BITS_DESC_s
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_RevPeekEntry_Clnt(RT_NETFIFO_PEEKENTRY_ARG_s *pArg,
int *pResult, int Clnt );
-void RT_NetFifo_RevPeekEntry_Svc(RT_NETFIFO_PEEKENTRY_ARG_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
-
-/**
- * [in] RT_NETFIFO_REMOVEENTRY_ARG_s
- * [out] int
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_RemoveEntry_Clnt(RT_NETFIFO_REMOVEENTRY_ARG_s *pArg,
int *pResult, int Clnt );
-void RT_NetFifo_RemoveEntry_Svc(RT_NETFIFO_REMOVEENTRY_ARG_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
-
-/**
- * [in] (RT_NETFIFO_HDLR_s *)
- * [out] RT_NETFIFO_BITS_DESC_s
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_PrepareEntry_Clnt(RT_NETFIFO_HDLR_s **pArg, int *pRe-
sult, int Clnt );
-void RT_NetFifo_PrepareEntry_Svc(RT_NETFIFO_HDLR_s **pArg, AMBA_IPC_SVC_RESULT_s *pRet);
-
-/**
- * [in] RT_NETFIFO_WRITEENTRY_ARG_s
- * [out] int
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_WriteEntry_Clnt(RT_NETFIFO_WRITEENTRY_ARG_s *pArg, int
*pResult, int Clnt );
-void RT_NetFifo_WriteEntry_Svc(RT_NETFIFO_WRITEENTRY_ARG_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
-
-/**
- * [in] void
- * [out] RT_NETFIFO_MEDIA_STREAMID_LIST_s
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_GetMediaStreamIDList_Clnt(void *pArg, int *pResult,
int Clnt );
-void RT_NetFifo_GetMediaStreamIDList_Svc(void *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
-
-/**
- * [in] int
- * [out] RT_NETFIFO_MOVIE_INFO_CFG_s
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_GetMediaInfo_Clnt(unsigned int *pArg, int *pResult,
int Clnt );
-void RT_NetFifo_GetMediaInfo_Svc(unsigned int *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
-
-/**
- * [in] RT_NETFIFO_PLAYBACK_OP_PARAM_s
- * [out] RT_NETFIFO_PLAYBACK_OP_PARAM_s
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_Playback_OP_Clnt(RT_NETFIFO_PLAYBACK_OP_PARAM_s *pArg,
int *pResult, int Clnt );
-void RT_NetFifo_Playback_OP_Svc(RT_NETFIFO_PLAYBACK_OP_PARAM_s *pArg, AMBA_IPC_SVC_
RESULT_s *pRet);
-
-/**
- * [in] unsigned int
- * [out] int
- */
-AMBA_IPC_REPLY_STATUS_e RT_NetFifo_ReportStatus_Clnt(unsigned int *pArg, int *pResult,
int Clnt );
-void RT_NetFifo_ReportStatus_Svc(unsigned int *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
-
-#endif /* _RPC_PROG_RT_NETFIFO_H_ */
+/**
+ * @file inc/mw/net/rpcprog/AmbaIPC_RpcProg_RT_NetFifo.h
+ *
+ * Header file for NetFifo RPC Services (RTOS side)
+ *
+ * Copyright (C) 2014, Ambarella, Inc.
+ *
+ * All rights reserved. No Part of this file may be reproduced, stored
+ * in a retrieval system, or transmitted, in any form, or by any means,
+ * electronic, mechanical, photocopying, recording, or otherwise,
+ * without the prior consent of Ambarella, Inc.
+ */
+#ifndef _RPC_PROG_RT_NETFIFO_H_
+#define _RPC_PROG_RT_NETFIFO_H_
+
+#include “AmbaIPC_Rpc_Def.h”
+
+#ifndef FOREVER
+#define FOREVER 0xFFFFFFFF
+#endif
+
+//RPC_INFO definition
+#define RT_NETFIFO_PROG_ID    0x10000004
+#define RT_NETFIFO_HOST AMBA_IPC_HOST_THREADX
+#define RT_NETFIFO_VER (1)
+#define RT_NETFIFO_DEFULT_TIMEOUT (FOREVER)
+#define RT_NETFIFO_NAME “LINK_RPC_SVC_NETFIFO”
+
+//===== mw.h =====
+#define RT_NETFIFO_EVENT_START_NUM (0x04000000) //Should be sync with AMP_FIFO_EVENT_
START_NUM in mw.h
+
+//===== fifo.h =====
+#define RT_NETFIFO_MARK_EOS          0x00FFFFFF
+#define RT_NETFIFO_MARK_EOS_PAUSE    0x00FFFFFE
+
+typedef enum _RT_NETFIFO_EVENT_e_ {
+    RT_NETFIFO_EVENT_DATA_CONSUMED = RT_NETFIFO_EVENT_START_NUM, /**< data consumed event
to data provider (write fifo)*/
+    RT_NETFIFO_EVENT_DATA_EOS, /**< data end of stream to data consumer (read fifo)*/
+    RT_NETFIFO_EVENT_DATA_READY, /**< new data ready to data consumer (read fifo)*/
+    RT_NETFIFO_EVENT_GET_WRITE_POINT, /**< get write pointer info from data consumer
(write fifo)*/
+    RT_NETFIFO_EVENT_RESET_FIFO /**< ask data consumer (write fifo) to reset everything
for bitsfifo reuse*/
+} RT_NETFIFO_EVENT_e;
+
+typedef enum _RT_NETFIFO_FRMAE_TYPE_e_ {
+    RT_NETFIFO_TYPE_MJPEG_FRAME = 0,    ///< MJPEG frame type
+    RT_NETFIFO_TYPE_IDR_FRAME = 1,    ///< Idr frame type
+    RT_NETFIFO_TYPE_I_FRAME = 2,      ///< I frame type
+    RT_NETFIFO_TYPE_P_FRAME = 3,      ///< P frame type
+    RT_NETFIFO_TYPE_B_FRAME = 4,      ///< B frame type
+    RT_NETFIFO_TYPE_JPEG_FRAME = 5,   ///< jpeg main frame
+    RT_NETFIFO_TYPE_THUMBNAIL_FRAME = 6,  ///< jpeg thumbnail frame
+    RT_NETFIFO_TYPE_SCREENNAIL_FRAME = 7, ///< jpeg screennail frame
+    RT_NETFIFO_TYPE_AUDIO_FRAME = 8,      ///< audio frame
+    RT_NETFIFO_TYPE_UNDEFINED = 9,        ///< others
+
+    RT_NETFIFO_TYPE_DECODE_MARK = 101, ///< used when feeding bitstream to dsp. will push
out all frame. */
+    RT_NETFIFO_TYPE_EOS = 255,                  ///< eos bits that feed to raw buffer
+
+    RT_NETFIFO_TYPE_LAST = RT_NETFIFO_TYPE_EOS
+} RT_NETFIFO_FRMAE_TYPE_e;
+
+typedef struct _RT_NETFIFO_s_ {
+    unsigned int nFifoId; /**< the unique of the fifo */
+    void *Ctx; /**< private data of the fifo */
+} RT_NETFIFO_HDLR_s;
+
+typedef struct _RT_NETFIFO_BITS_DESC_s_ {
+    unsigned int SeqNum; /**< sequential number of bits buffer */
+    unsigned long long Pts; /**< time stamp in ticks */
+    unsigned char Type; /**< data type of the entry. see RT_NETFIFO_FRMAE_TYPE_e*/
+    unsigned char Completed; /**< if the buffer content a complete entry */
+    unsigned short  Align; /** data size alignment (in bytes, align = 2^n, n is a integer
)*/
+    unsigned char* StartAddr; /**< start address of data */
+    unsigned int Size; /**< real data size */
+} RT_NETFIFO_BITS_DESC_s;
+
+
+typedef enum _RT_NETFIFO_CFG_INIT_DATA_FETCH_CONDITION_e_{
+    RT_NETFIFO_CFG_INIT_DISABLE = 0,  /**< no neeed for init data */
+    RT_NETFIFO_CFG_INIT_WITH_TIME, /**< get frames with given length */
+    RT_NETFIFO_CFG_INIT_WITH_NUM_FRAME, /**< get frames with given frame number. ex. if
you set  NumFrame to 100, you will get 100 frames(if there are 100 frames)*/
+} RT_NETFIFO_CFG_INIT_DATA_FETCH_CONDITION_e;
+
+typedef struct _RT_NETFIFO_CFG_INIT_DATA_s_ {
+    unsigned char CreateFifoWithInitData; /**< If create fifo with init data (if data val-
id) */
+
+    RT_NETFIFO_CFG_INIT_DATA_FETCH_CONDITION_e InitCondition; /**< define how to descript
initial data required. */
+
+    union {
+        unsigned long long TimeLength; /**< In normal case, the value of backward fetch
is 0.\n
+         * it means fifo will output frames only after fifo is created.\n
+         * As the valus is not 0, fifo will try to find out the longest valid frames
+         */
+        unsigned long long NumFrame; /**< number of frame */
+    } InitParam; /**< parameter for determinate initial data*/
+
+    RT_NETFIFO_FRMAE_TYPE_e FristFrameType; /**< Frame type of first frame for valid data,
used on video stream to ensure start with Idr. */
+
+    unsigned long long OnCreateFirstFramePts; /** [OUT] Pts of first frame,
0xFFFFFFFFFFFFFFFF if first frame is not valid on create */
+    unsigned long long OnCreateTimeLength; /** [OUT] data length in fifo on create in ms
*/
+} RT_NETFIFO_CFG_INIT_DATA_s;
+
+typedef struct _RT_NETFIFO_CFG_s_ {
+    void *hCodec; /**< the codec the fifo is working on. */
+    unsigned int NumEntries; /**< Number of entries of a fifo */
+    unsigned int IsVirtual; /**< Creating virtual fifo (for data flow) or not (for codec)
*/
+    unsigned int cbEvent; /**< the callback function for fifo event */
+    unsigned int EventDataConsumedThreshold; /**<
+      * the threshold for event RT_NETFIFO_CALLBACK_EVENT_DATA_CONSUMED\n
+      * event only triggered if remain data us under the threshold.
+      * 0 to disable
+      */
+    unsigned int EventDataReadySkipNum; /**<
+      * if not 0, RT_NETFIFO_CALLBACK_EVENT_DATA_READY will not be triggered every frame
encoded.\n
+      * it will be triggered every (eventDataReadySkipNum frame+1) frames
+      */
+    unsigned int cbGetWritePoint; /**<
+     * the callback function invoked when we receive prepareSpace.\n
+     * it should ONLY be used on fifo linked to codec \n
+     * it only take effect on a codec read fifo.\n
+     * the callback should be registered by a decode mgr for all decode codec\n
+     */
+
+    unsigned char SyncRpOnWrite;
+    /**< if 0, rp of base fifo will be updated on remove of virtual fifo.\n
+     *   if 1, rp of base fifo will be updated on write of base fifo.\n
+     *   ONLY take effect on base fifo.
+     */
+
+    unsigned int RawBaseAddr; /**< when sync rp on write is on and RawBaseAddr/RawLimitA-
ddr is given, base fifo will remove desc automatically on overwrite. \n
+                        * only needed on base fifo and sync on write is on
+                        */
+
+    unsigned int RawLimitAddr; /**< when sync rp on write is on and RawBaseAddr/RawLim-
itAddr is given, base fifo will remove desc automatically on overwrite. \n
+                        * only needed on base fifo and sync on write is on
+                        */
+
+    unsigned long long TickPerSecond; /**< TiackPerSecond used for Pts for the fifo. \n
+    Could be 0 if we whould NOT like to use init data. */
+
+    RT_NETFIFO_CFG_INIT_DATA_s InitData; /**< init status for virtual fifo */
+} RT_NETFIFO_CFG_s;
+
+typedef struct _RT_NETFIFO_INFO_s_ {
+    unsigned int TotalEntries; /**< total number of entries */
+    unsigned int AvailEntries; /**< number of entries with data*/
+} RT_NETFIFO_INFO_s;
+
+//===== NetFifo.h(Muxer.h) =====
+typedef enum _RT_NETFIFO_MEDIA_TRACK_TYPE_e_ {
+    RT_NETFIFO_MEDIA_TRACK_TYPE_VIDEO = 0x01,  /**< The track’s type is Video */
+    RT_NETFIFO_MEDIA_TRACK_TYPE_AUDIO = 0x02,  /**< The track’s type is Audio */
+    RT_NETFIFO_MEDIA_TRACK_TYPE_TEXT = 0x03,   /**< The track’s type is Text */
+    RT_NETFIFO_MEDIA_TRACK_TYPE_MAX = 0x04     /**< Max value, for check use */
+} RT_NETFIFO_MEDIA_TRACK_TYPE_e;
+
+typedef struct _RT_NETFIFO_VIDEO_TRACK_CFG_s_ {
+    unsigned int nGOPSize;             /**< The number of the picture between IDR pic-
tures. */
+    unsigned int nTrickRecDen;         /**< For AmpFormat_ConvertPTS(), the Denominator
of the trick record. If is high frame rate, the denominator and numerator are specify the
factor than default 30fps, Ex: 120fps -> Den = 1, Num = 4 */
+    unsigned int nTrickRecNum;         /**< For AmpFormat_ConvertPTS(), the Numerator of
the trick record. If is high frame rate, the denominator and numerator are specify the
factor than default 30fps, Ex: 120fps -> Den = 1, Num = 4 */
+    unsigned int nCodecTimeScale;     /**< TODO: It needs get from H264 bitstream, it is
not configurable. */
+    unsigned short  nWidth;               /**< Picture width */
+    unsigned short  nHeight;              /**< Picture height */
+    unsigned short  nM;                   /**< The number of the picture between reference
pictures(IDR, I, P) */
```

```
+    unsigned short  nN;                   /**< The number of the picture between I pic-
tures */
+    unsigned char bDefault;              /**< The flag defines the track as default video
track, if the media hasn't one video track. */
+    unsigned char nMode;                 /**< The value defines the picture mode of the
video. It has progressive and interlaced mode. Interlaced mode has Field Per Sample and
Frame Per Sample, See AMP_VIDEO_MODE_s */
+    unsigned char bClosedGOP;            /**< The structure of the Close GOP is I P B B P
B B. The structure of the Open GOP is I B B P B B, If resume or auto split, the value al-
ways is Open GOP. */
+} RT_NETFIFO_VIDEO_TRACK_CFG_s;
+
+typedef struct _RT_NETFIFO_AUDIO_TRACK_CFG_s_ {
+    unsigned int nSampleRate;          /**< The sample rate(Hz) of the audio track. */
+    unsigned char bDefault;              /**< The flag defines the track as default audio
track, if the media hasn't one audio track. */
+    unsigned char nChannels;             /**< The number of audio channel. */
+    unsigned char nBitsPerSample;        /**< The per sample size of the audio track. Ex:
8 bits, 16 bits, ....n bits */
+} RT_NETFIFO_AUDIO_TRACK_CFG_s;
+
+typedef struct _RT_NETFIFO_TEXT_TRACK_CFG_s_ {
+    unsigned char bDefault;              /**< The flag defines the track as default text
track, if the media hasn't one text track. */
+} RT_NETFIFO_TEXT_TRACK_CFG_s;
+
+typedef struct _RT_NETFIFO_MEDIA_TRACK_CFG_s_{
+    unsigned int nMediaId;             /**< The media type of the track. The id is media
id, See AMP_FORMAT_MID_e */
+    unsigned int nTimeScale;           /**< Time scale, the same as the LCM of those of
all tracks */
+    unsigned int nTimePerFrame;        /**< The time of the frame that the unit of it is
Time scale. */
+    unsigned int nInitDelay;           /**< Initial delay time(ms) of the track. */
+    void *hCodec;             /**< the codec which this track is working on. */
+    unsigned char *pBufferBase;             /**< The start address of the FIFO of the
track. User pushs data to the FIFO, the FIFO will write data to the address of the buffer.
*/
+    unsigned char *pBufferLimit;            /**< The end address of the FIFO of the track,
The data can't write overlap the address, FIFO size = FIFO buffer limit - FIFO buffer
base. */
+    union {
+        RT_NETFIFO_VIDEO_TRACK_CFG_s Video;
+        RT_NETFIFO_AUDIO_TRACK_CFG_s Audio;
+        RT_NETFIFO_TEXT_TRACK_CFG_s Text;
+    } Info;
+    unsigned char nTrackType;            /**< Track type, See AMP_MEDIA_TRACK_TYPE_e */
+} RT_NETFIFO_MEDIA_TRACK_CFG_s;
+
+#define RT_NETFIFO_MAX_TRACK_PER_MEDIA  4   /**< The max track number of a media. */
+
+typedef struct _RT_NETFIFO_MOVIE_INFO_CFG_s_ {
+    RT_NETFIFO_MEDIA_TRACK_CFG_s Track[RT_NETFIFO_MAX_TRACK_PER_MEDIA];
+    unsigned char nTrack;                /**< The number of Tracks in the movie. */
+} RT_NETFIFO_MOVIE_INFO_CFG_s;
+
+//===== self sepcific =====
+typedef struct _RT_NETFIFO_PEEKENTRY_ARG_s_ {
```

```
+    RT_NETFIFO_HDLR_s *fifo;
+    unsigned int distanceToLastEntry;
+} RT_NETFIFO_PEEKENTRY_ARG_s;
+
+typedef struct _RT_NETFIFO_REMOVEENTRY_ARG_s_ {
+    RT_NETFIFO_HDLR_s *fifo;
+    unsigned int EntriesToBeRemoved;
+} RT_NETFIFO_REMOVEENTRY_ARG_s;
+
+typedef struct _RT_NETFIFO_WRITEENTRY_ARG_s_ {
+    RT_NETFIFO_HDLR_s *fifo;
+    RT_NETFIFO_BITS_DESC_s desc;
+} RT_NETFIFO_WRITEENTRY_ARG_s;
+
+typedef struct _RT_NETFIFO_MEDIA_STREAMID_LIST_s_ {
+    int Amount;
+    int StreamID_List[16];
+} RT_NETFIFO_MEDIA_STREAMID_LIST_s;
+
+typedef struct _RT_NETFIFO_PLAYBACK_OP_PARAM_s_ {
+    unsigned int OP;
+    unsigned char Param[128];
+} RT_NETFIFO_PLAYBACK_OP_PARAM_s;
+
+//============ RPC_FUNC definition ============
+enum _RT_NETFIFO_FUNC_e_ {
+    RT_NETFIFO_FUNC_GETDEFAULTCFG = 1,
+    RT_NETFIFO_FUNC_CREATE,
+    RT_NETFIFO_FUNC_DELETE,
+    RT_NETFIFO_FUNC_ERASEALL,
+    RT_NETFIFO_FUNC_GETINFO,
+    RT_NETFIFO_FUNC_PEEKENTRY,
+    RT_NETFIFO_FUNC_REVPEEKENTRY,
+    RT_NETFIFO_FUNC_REMOVEENTRY,
+    RT_NETFIFO_FUNC_PREPAREENTRY,
+    RT_NETFIFO_FUNC_WRITEENTRY,
+    RT_NETFIFO_FUNC_GETMEDIASTREAMID,
+    RT_NETFIFO_FUNC_GETMEDIAINFO,
+    RT_NETFIFO_FUNC_PLAYBACK_OP,
+    RT_NETFIFO_FUNC_REPORTSTATUS,
+
+    RT_NETFIFO_FUNC_AMOUNT
+};
+
+
+/**
+ * [in] NULL
+ * [out] RT_NETFIFO_CFG_s
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_GetDefaultCfg_Clnt(void *pArg, int *pResult, int Clnt
);
+void RT_NetFifo_GetDefaultCfg_Svc(void *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
+
+/**
+ * [in] RT_NETFIFO_CFG_s
+ * [out] (RT_NETFIFO_HDLR_s**)
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_Create_Clnt(RT_NETFIFO_CFG_s *pArg, int *pResult, int
Clnt );
+void RT_NetFifo_Create_Svc(RT_NETFIFO_CFG_s *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
+
+/**
+ * [in] (RT_NETFIFO_HDLR_s *)
+ * [out] int
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_Delete_Clnt(RT_NETFIFO_HDLR_s **pArg, int *pResult,
int Clnt );
+void RT_NetFifo_Delete_Svc(RT_NETFIFO_HDLR_s **pArg, AMBA_IPC_SVC_RESULT_s *pRet);
+
+/**
+ * [in] (RT_NETFIFO_HDLR_s *)
+ * [out] int
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_EraseAll_Clnt(RT_NETFIFO_HDLR_s **pArg, int *pResult,
int Clnt );
+void RT_NetFifo_EraseAll_Svc(RT_NETFIFO_HDLR_s **pArg, AMBA_IPC_SVC_RESULT_s *pRet);
+
+/**
+ * [in] (RT_NETFIFO_HDLR_s *)
+ * [out] RT_NETFIFO_INFO_s
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_GetInfo_Clnt(RT_NETFIFO_HDLR_s **pArg, int *pResult,
int Clnt );
+void RT_NetFifo_GetInfo_Svc(RT_NETFIFO_HDLR_s **pArg, AMBA_IPC_SVC_RESULT_s *pRet);
+
+/**
+ * [in] RT_NETFIFO_PEEKENTRY_ARG_s
+ * [out] RT_NETFIFO_BITS_DESC_s
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_PeekEntry_Clnt(RT_NETFIFO_PEEKENTRY_ARG_s *pArg, int
*pResult, int Clnt );
+void RT_NetFifo_PeekEntry_Svc(RT_NETFIFO_PEEKENTRY_ARG_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
+
+/**
+ * [in] RT_NETFIFO_PEEKENTRY_ARG_s
+ * [out] RT_NETFIFO_BITS_DESC_s
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_RevPeekEntry_Clnt(RT_NETFIFO_PEEKENTRY_ARG_s *pArg,
int *pResult, int Clnt );
+void RT_NetFifo_RevPeekEntry_Svc(RT_NETFIFO_PEEKENTRY_ARG_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
+
+/**
+ * [in] RT_NETFIFO_REMOVEENTRY_ARG_s
+ * [out] int
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_RemoveEntry_Clnt(RT_NETFIFO_REMOVEENTRY_ARG_s *pArg,
int *pResult, int Clnt );
+void RT_NetFifo_RemoveEntry_Svc(RT_NETFIFO_REMOVEENTRY_ARG_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
+
+/**
```

```
+ * [in] (RT_NETFIFO_HDLR_s *)
+ * [out] RT_NETFIFO_BITS_DESC_s
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_PrepareEntry_Clnt(RT_NETFIFO_HDLR_s **pArg, int *pRe-
sult, int Clnt );
+void RT_NetFifo_PrepareEntry_Svc(RT_NETFIFO_HDLR_s **pArg, AMBA_IPC_SVC_RESULT_s *pRet);
+
+/**
+ * [in] RT_NETFIFO_WRITEENTRY_ARG_s
+ * [out] int
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_WriteEntry_Clnt(RT_NETFIFO_WRITEENTRY_ARG_s *pArg, int
*pResult, int Clnt );
+void RT_NetFifo_WriteEntry_Svc(RT_NETFIFO_WRITEENTRY_ARG_s *pArg, AMBA_IPC_SVC_RESULT_s
*pRet);
+
+/**
+ * [in] void
+ * [out] RT_NETFIFO_MEDIA_STREAMID_LIST_s
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_GetMediaStreamIDList_Clnt(void *pArg, int *pResult,
int Clnt );
+void RT_NetFifo_GetMediaStreamIDList_Svc(void *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
+
+/**
+ * [in] int
+ * [out] RT_NETFIFO_MOVIE_INFO_CFG_s
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_GetMediaInfo_Clnt(unsigned int *pArg, int *pResult,
int Clnt );
+void RT_NetFifo_GetMediaInfo_Svc(unsigned int *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
+
+/**
+ * [in] RT_NETFIFO_PLAYBACK_OP_PARAM_s
+ * [out] RT_NETFIFO_PLAYBACK_OP_PARAM_s
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_Playback_OP_Clnt(RT_NETFIFO_PLAYBACK_OP_PARAM_s *pArg,
int *pResult, int Clnt );
+void RT_NetFifo_Playback_OP_Svc(RT_NETFIFO_PLAYBACK_OP_PARAM_s *pArg, AMBA_IPC_SVC_
RESULT_s *pRet);
+
+/**
+ * [in] unsigned int
+ * [out] int
+ */
+AMBA_IPC_REPLY_STATUS_e RT_NetFifo_ReportStatus_Clnt(unsigned int *pArg, int *pResult,
int Clnt );
+void RT_NetFifo_ReportStatus_Svc(unsigned int *pArg, AMBA_IPC_SVC_RESULT_s *pRet);
+
+#endif /* _RPC_PROG_RT_NETFIFO_H_ */
```

A1.2 Data Flow

```
[diff] A12SDK/rtos/vendors/ambarella/inc/mw/cfs/AmpCfs.h

@@ -1,11 +1,6 @@
 /**
  * @file AmpCfs.h
  *
- * History:
- *    2013/05/22 - [Irene Wang] created file
- *    2013/09/25 - [Yi-Ching Liao] modified
- *
- *
  * Copyright (C) 2004-2013, Ambarella, Inc.
  *
  * All rights reserved. No Part of this file may be reproduced, stored

[Purpose]
  Remove useless comment

-------------------------------------------------

@@ -176,8 +171,7 @@ typedef struct AMP_CFS_CFG_s_ {
     int (*FileOperation)(int, UINT32);                      /**< The callback function to
report all events of file operations in the CFS module */
     UINT32 CacheMaxFileNum;                                 /**< The maximum number of
cached files */
     UINT32 SchBankSize;                                     /**< The size of a bank */
-    UINT32 SchTaskPriority;                                 /**< Task priority */
-    UINT32 SchStackSize;                                    /**< Stack size of tasks */
+    AMP_TASK_INFO_s TaskInfo;                               /**< The information of CFS
background task */
     UINT32 SchLowTxRate[AMP_CFS_MAX_TASK_AMOUNT];           /**< The minimum acceptable
transmission rate (Its unit is KBps. It is for AsyncMode.) */
     UINT32 SchLowSpeedSize[AMP_CFS_MAX_TASK_AMOUNT];        /**< The tolerance size of
writing data continually in low speed */
     UINT8  SchTaskDriveTable[AMP_CFS_MAX_DRIVE_AMOUNT];     /**< The mapping table used
to convert drive name to task ID */

[Purpose]
  Add TaskInfo

-------------------------------------------------

@@ -249,6 +243,15 @@ extern int AmpCFS_Init(AMP_CFS_CFG_s *config);
  * Stream level
  */
 /**
+ *  Get the max cached data size of a file
+ *  @param [in] file The file descriptor
+ *  @param [out] dataSize The returned data size (bytes)
+ *  @return 0 - OK, others - AMP_ER_CODE_e
+ *  @see AMP_ER_CODE_e
+ */
+int AmpCFS_GetFileMaxCachedDataSize(AMP_CFS_FILE_s *file, UINT64 *dataSize);
+
+/**
  *  Get default file parameters for opening a file descriptor.
  *  @param [out] fileParam The returned file parameters
  *  @return 0 - OK, others - AMP_ER_CODE_e
```

[Purpose]
  Add AmpCFS_GetFileMaxCachedDataSize

```
@@ -502,6 +505,22 @@ extern int AmpCFS_Format(char driveName, const char *param);
  */
 extern int AmpCFS_Sync(char driveName, int mode);

+/**
+ *  Mount a drive
+ *  @param [in] driveName Drive name (from A to Z)
+ *  @return 0 - AMP_OK, others - AMP_ER_CODE_e
+ *  @see AMP_ER_CODE_e
+ */
+extern int AmpCFS_Mount(char driveName);
+
+/**
+ *  Unmount a drive
+ *  @param [in] driveName Drive name (from A to Z)
+ *  @return 0 - AMP_OK, others - AMP_ER_CODE_e
+ *  @see AMP_ER_CODE_e
+ */
+extern int AmpCFS_Unmount(char driveName);
+
 /*
  * Cache Operations
  */
```

[Purpose]
  Add CFS mount & unmount

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/dcf/AmpDcf.h

```
@@ -3,10 +3,6 @@
  *
  *  DCF header
  *
- *    2012/10/22 - [Irene Wang] created file
- *    2013/09/30 - [Chen-Lung Chan] refined
- *
- *
  * Copyright (C) 2013, Ambarella, Inc.
  *
  * All rights reserved. No Part of this file may be reproduced, stored
```

[Purpose]

```
  Remove useless comment

--------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/Demuxer.h

@@ -3,12 +3,6 @@
  *
  *  Demuxer header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/05/07 |clchan      |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

--------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/DummyMux.h

@@ -3,12 +3,6 @@
  *
  *  DUMMY Muxing Format header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/04/16 |clchan      |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

--------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/Editor.h

@@ -3,12 +3,6 @@
  *
  *  Editor header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/09/25 |felix       |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

--------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/Exif.h

@@ -3,12 +3,6 @@
  *
  *  Common definitions for Exif format
  *
- *  **History**
- *      |Date        |Name        |Comments        |
- *      |------------|------------|----------------|
- *      |2013/05/31  |yhlee       |Created         |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *   No part of this file may be reproduced, stored in a retrieval system,
  *   or transmitted, in any form, or by any means, electronic, mechanical, photocopying,

[Purpose]
  Remove useless comment

--------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/ExifDmx.h

@@ -3,12 +3,6 @@
  *
  *  Exif Demuxing Format header
  *
- *  **History**
- *      |Date        |Name        |Comments        |
- *      |------------|------------|----------------|
- *      |2013/06/17  |yhlee       |Created         |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *   No part of this file may be reproduced, stored in a retrieval system,
  *   or transmitted, in any form, or by any means, electronic, mechanical, photocopying,

[Purpose]
  Remove useless comment

--------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/ExifMux.h

@@ -3,12 +3,6 @@
```

```
  *
  *  Exif Muxing Format header
  *
- *  **History**
- *      |Date        |Name         |Comments        |
- *      |------------|-------------|----------------|
- *      |2013/05/10  |yhlee        |Created         |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *   No part of this file may be reproduced, stored in a retrieval system,
  *   or transmitted, in any form, or by any means, electronic, mechanical, photocopying,
```

[Purpose]
  Remove useless comment

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/Ext.h

```
@@ -3,12 +3,6 @@
  *
  *  Common definitions for External format
  *
- *  **History**
- *      |Date        |Name         |Comments        |
- *      |------------|-------------|----------------|
- *      |2013/07/17  |clchan       |Created         |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                  No part of this file may be reproduced, stored in a retrieval system,
  *                  or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,
```

[Purpose]
  Remove useless comment

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/ExtDmx.h

```
@@ -3,12 +3,6 @@
  *
  *  External Demuxing Format header
  *
- *  **History**
- *      |Date        |Name         |Comments        |
- *      |------------|-------------|----------------|
- *      |2013/05/14  |clchan       |Created         |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                  No part of this file may be reproduced, stored in a retrieval system,
  *                  or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,
[Purpose]
  Remove useless comment

-------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/ExtMux.h

@@ -3,12 +3,6 @@
  *
  *  External Muxing Format header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |------------|-------------|---------------|
- *      |2013/05/14 |clchan      |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

-------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/Format.h

@@ -3,12 +3,6 @@
  *
  *  Format common function header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |------------|-------------|---------------|
- *      |2013/03/28 |clchan      |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

-------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/FormatDef.h

@@ -3,12 +3,6 @@
  *
  *  Format definition header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/07/17 |clchan      |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,
```

[Purpose]
  Remove useless comment

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/Iso.h

```
@@ -3,12 +3,6 @@
  *
  *  Common definitions for ISO format
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/08/22 |felix       |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,
```

[Purpose]
  Remove useless comment

```
@@ -106,6 +100,7 @@ typedef struct {
     ISO_TRACK_INFO_s TrackInfo[AMP_FORMAT_MAX_TRACK_PER_MEDIA]; /**< The track informa-
tion in the private data (See ISO_TRACK_INFO_s.) */
     UINT64 FrameDataSize;
     UINT64 FrameDataPos;
+    UINT64 MaxCachedSize;
     BOOL8 EnableCO64;
 } AMP_ISO_PRIV_INFO_s;
```

[Purpose]
  Add max cached size parameters.

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/Matroska.h

```
@@ -3,12 +3,6 @@
  *
  *  Common definitions of Matroska format
  *
```

```
- *  **History**
- *      |Date       |Name        |Comments        |
- *      |-----------|------------|----------------|
- *      |2014/05/15 |clchan      |Created         |
- *
- *
  *  @copyright 2014 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,
```

[Purpose]
  Remove useless comment

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/MkvDmx.h

```
@@ -3,12 +3,6 @@
  *
  *  MKV Demuxing Format header
  *
- *  **History**
- *      |Date       |Name        |Comments        |
- *      |-----------|------------|----------------|
- *      |2014/10/17 |clchan      |Created         |
- *
- *
  *  @copyright 2014 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,
```

[Purpose]
  Remove useless comment

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/MkvEdt.h

```
@@ -3,12 +3,6 @@
  *
  *  MKV Editing format module header
  *
- *  **History**
- *      |Date       |Name        |Comments        |
- *      |-----------|------------|----------------|
- *      |2014/11/26 |clchan      |Created         |
- *
- *
  *  @copyright 2014 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,
```

[Purpose]

```
  Remove useless comment

-------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/MkvMux.h

@@ -3,12 +3,6 @@
  *
  *  MKV Muxing Format header
  *
- *  **History**
- *      |Date       |Name        |Comments        |
- *      |-----------|------------|----------------|
- *      |2014/02/10 |clchan      |Created         |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

-------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/MovDmx.h

@@ -3,12 +3,6 @@
  *
  *  Mov Demuxing Format header
  *
- *  **History**
- *      |Date       |Name        |Comments        |
- *      |-----------|------------|----------------|
- *      |2013/08/21 |felix       |Created         |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

-------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/MovEdt.h

@@ -3,12 +3,6 @@
  *
  *  MOV Editing format module header
  *
- *  **History**
- *      |Date       |Name        |Comments        |
- *      |-----------|------------|----------------|
- *      |2015/01/28 |clchan      |Created         |
- *
- *
  *  @copyright 2015 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

--------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/MovMux.h

@@ -3,12 +3,6 @@
  *
  *  Mov Muxing Format header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/08/21 |felix       |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

--------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/Mp4Dmx.h

@@ -3,12 +3,6 @@
  *
  *  MP4 Demuxing Format header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/06/06 |felix       |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

--------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/Mp4Edt.h
```

```
@@ -3,12 +3,6 @@
  *
  *  MP4 Editing format module header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/09/26 |felix       |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

-------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/Mp4Mux.h

@@ -3,12 +3,6 @@
  *
  *  Mp4 Muxing Format header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/05/15 |felix       |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

-------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/Muxer.h

@@ -3,12 +3,6 @@
  *
  *  Muxer header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/03/28 |clchan      |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,
[Purpose]
  Remove useless comment

--------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/index/Index.h

@@ -3,12 +3,6 @@
  *
  *  encode common function header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/05/13 |felix       |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

--------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/index/Mem.h

@@ -3,12 +3,6 @@
  *
  *  encode common function header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/12/09 |felix       |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

--------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/index/Raw.h

@@ -3,12 +3,6 @@
  *
  *  encode common function header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/12/02 |felix       |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

-------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/index/Temp.h

@@ -3,12 +3,6 @@
  *
  *  encode common function header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/05/13 |felix       |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

-------------------------------------------------

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/stream/File.h

@@ -3,12 +3,6 @@
  *
  *  FILE stream header
  *
- *  **History**
- *      |Date       |Name        |Comments       |
- *      |-----------|------------|---------------|
- *      |2013/04/16 |clchan      |Created        |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,

[Purpose]
  Remove useless comment

-------------------------------------------------
```

[diff] A12SDK/rtos/vendors/ambarella/inc/mw/format/stream/Stream.h

```
@@ -3,12 +3,6 @@
  *
  *  encode common function header
  *
- *  **History**
- *      |Date       |Name         |Comments        |
- *      |-----------|-------------|----------------|
- *      |2013/03/29 |clchan       |Created         |
- *
- *
  *  @copyright 2013 Ambarella Corporation. All rights reserved.
  *                 No part of this file may be reproduced, stored in a retrieval system,
  *                 or transmitted, in any form, or by any means, electronic, mechanical,
photocopying,
```

[Purpose]
  Remove useless comment

```
@@ -31,7 +25,9 @@

 #include <format/FormatDef.h>

-#define AMP_STREAM_OP_INSERT_SPACE  0x00000001      /**< The opcode of the function to in-
sert spaces at a specified position */
+#define AMP_STREAM_OP_INSERT_SPACE  0x00000001          /**< The opcode of the function to
insert spaces at a specified position */
+#define AMP_STREAM_OP_GET_MAX_CACHED_SIZE  0x00000002   /**< The opcode of the function to
get cached data size*/
+
 // for external muxer/demuxer
 #define AMP_STREAM_OP_SET_MEDIA_INFO    0x00000010  /**< The opcode of the function to set
media information */
 #define AMP_STREAM_OP_SET_FRAME_INFO    0x00000011  /**< The opcode of the function to set
frame information */
```

[Purpose]
  Add max cached size parameters.

Appendix 2 Additional Resources

Please contact an Ambarella representative for digital copies.

- *System*
  - *AMBARELLA_SDK6_API_AmbaKAL*
  - *AMBARELLA_SDK6_API_B5*
  - *AMBARELLA_SDK6_API_System*
  - *AMBARELLA_SDK6_API_AmbaFS*
  - *AMBARELLA_SDK6_AN_DRAM_Tuning*
  - *AMBARELLA_SDK6_AN_ADC_And_IR_Input*
  - *AMBARELLA_SDK6_AN_USB*
  - *AMBARELLA_SDK6_AN_Build_Environment*

- *AmbaLink*
  - *AMBARELLA_SDK6_API_AmbaLink*
  - *AMBARELLA_SDK6_AN_AmbaLink*

- *Driver*
  - *AMBARELLA_SDK6_AN_Custom_LCD_Driver*
  - *AMBARELLA_SDK6_AN_Custom_Image_Sensor_Driver*
  - *AMBARELLA_SDK6_AN_Custom_Audio CODEC_Driver*

- *Calibration*
  - *AMBARELLA_SDK6_UG_Calibration*
  - *AMBARELLA_SDK6_API_Calibration*

- *Image Quality*
  - *AMBARELLA_SDK6_UG_A9_Amage (AMBARELLA_SDK6_UG_A12_Amage)*
  - *AMBARELLA_SDK6_AN_A9_IQ_Tuning (AMBARELLA_SDK6_AN_A12_IQ_Tuning)*
  - *AMBARELLA_SDK6_API_Image_Processing*

- *Middleware*
  - *AMBARELLA_SDK6_UG_MW_UnitTest*
  - *AMBARELLA_SDK6_API_Middleware_Service*
  - *AMBARELLA_SDK6_API_Middleware_Flow*

- *Connected App*
    - *AMBARELLA_SDK6_API_Connected_AppLib*
    - *AMBARELLA_SDK6_DS_Connected_Application*

Appendix 3 Important Notice

All Ambarella design specifications, datasheets, drawings, files, and other documents (together and separately, “materials”) are provided on an “*as is*” basis, and Ambarella makes no warranties, expressed, implied, statutory, or otherwise with respect to the materials, and expressly disclaims all implied warranties of noninfringement, merchantability, and fitness for a particular purpose. The information contained herein is believed to be accurate and reliable. However, Ambarella assumes no responsibility for the consequences of use of such information.

Ambarella Incorporated reserves the right to correct, modify, enhance, improve, and otherwise change its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

All products are sold subject to Ambarella’s terms and conditions of sale supplied at the time of order acknowledgment. Ambarella warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with its standard warranty. Testing and other quality control techniques are used to the extent Ambarella deems necessary to support this warranty.

Ambarella assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using Ambarella components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

Ambarella does not warrant or represent that any license, either expressed or implied, is granted under any Ambarella patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which Ambarella products or services are used. Information published by Ambarella regarding third-party products or services does not constitute a license from Ambarella to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from Ambarella under the patents or other intellectual property of Ambarella.

Reproduction of information from Ambarella documents is not permissible without prior approval from Ambarella.

Ambarella products are not authorized for use in safety-critical applications (such as life support) where a failure of the product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Customers acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of Ambarella products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by Ambarella. Further, Customers must fully indemnify Ambarella and its representatives against any damages arising out of the use of Ambarella products in such safety-critical applications.

Ambarella products are neither designed nor intended for use in automotive and military/aerospace applications or environments. Customers acknowledge and agree that any such use of Ambarella products is solely at the Customer’s risk, and they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

Appendix 4 Revision History

NOTE: Page numbers for previous drafts may differ from page numbers in the current version.

Version	Date	Comments
1.0	30 June 2015	Draft Original
1.1	8 July 2015	Remove 1080p100 support for IMX206

Table A4-1. Revision History.