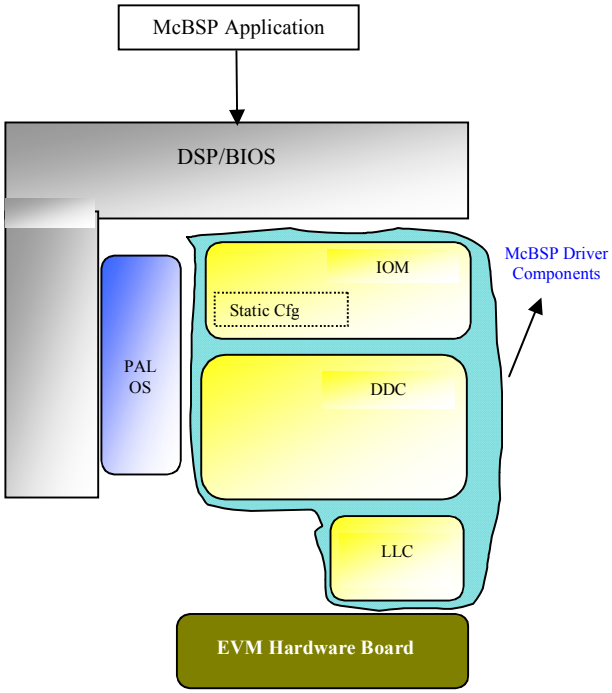
Features Supported by the DSP/BIOS McBSP driver

- Middle layer of the driver conforms to IO Mini driver model defined by the DSP/BIOS DDK (Device Driver Development Kit) Framework.
- Designed for (but not limited to) use with codec drivers.
- Keeps External Frame Sync.
- Multi-instance (Handles multiple serial ports simultaneously).
- Supports run-time Start/Stop of the Audio Play and Record operation.
- Supports Pause-Resume feature for Audio Playback operation when integrated with the audio codec specific driver.
- Supports Mute ON/OFF feature for Audio Playback operation when integrated with the audio codec specific driver.



Description

Details about the tools and the BIOS version that the driver is compatible with can be found in the system Release Notes.

The components used to develop the McBSP driver are given below.

1. **CSLR Modules used**
   - McBSP module.
2. **CPU Interrupts used**
   - EDMA transfer completion interrupt
3. **Peripherals Used**
   - McBSP

- EDMA
- EMIF for external memory interface

4. Drivers used

- EDMA

5. Modes supported by the driver(Data access mechanism)

- DMA interrupt mode.

The device independent layer of the McBSP driver is a class driver. The device independent layer of the driver is responsible for buffer management and application synchronization. The IO-mini layer contains the device-specific portions of the driver. From IOM mini driver DDC layer APIs are called from which LLC layer APIs are called.

The following table gives a quick overview of the supported class driver API services.

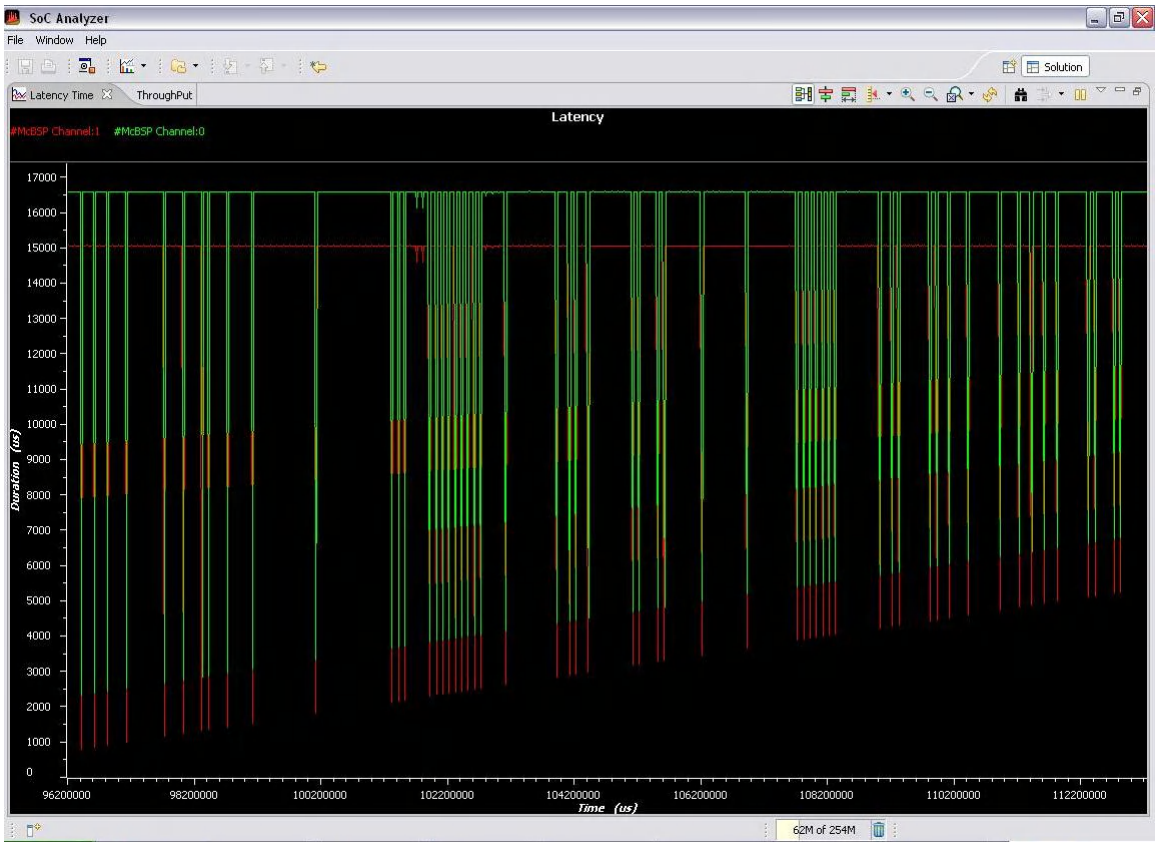
GIO_create()	Opens the McBSP driver for operation
GIO_delete(gioChan)	Closes the McBSP driver from operation.
GIO_submit()	Submits a request Read/write to McBSP driver
GIO_read()	Read data from McBSP driver.
GIO_write()	Write data to McBSP driver for transmit.
GIO_control()	Performs device specific control operations on the device. Supported control commands include: PSP_CTRL_McBSP_STOP PSP_CTRL_McBSP_START PSP_CTRL_McBSP_LOOPBACK PSP_CTRL_McBSP_PAUSE PSP_CTRL_McBSP_RESUME PSP_CTRL_McBSP_MUTE_ON PSP_CTRL_McBSP_MUTE_OFF PSP_CTRL_McBSP_CHAN_RESET PSP_CTRL_McBSP_DEVICE_RESET PSP_CTRL_McBSP_CNG_ADDR PSP_CTRL_McBSP_CONFIG_DATA PSP_CTRL_McBSP_SRGR_START PSP_CTRL_McBSP_SRGR_STOP

	PSP_CTRL_McBSP_FSGR_START PSP_CTRL_McBSP_FSGR_STOP PSP_CTRL_McBSP_SET_CLKMODE PSP_CTRL_McBSP_SET_FRMSYNCMODE PSP_CTRL_McBSP_CONFIG_SRGR PSP_CTRL_McBSP_SET_BCLK_POL PSP_CTRL_McBSP_SET_FRMSYNC_POL PSP_CTRL_McBSP_MODIFY_LOOPJOB PSP_CTRL_McBSP_RECEIVE_SYNCERR_INT_ENABLE PSP_CTRL_McBSP_XMIT_SYNCERR_INT_ENABLE
GIO_Flush ()	Flush the data in the channel.
GIO_Abort()	Abort the channel operation.

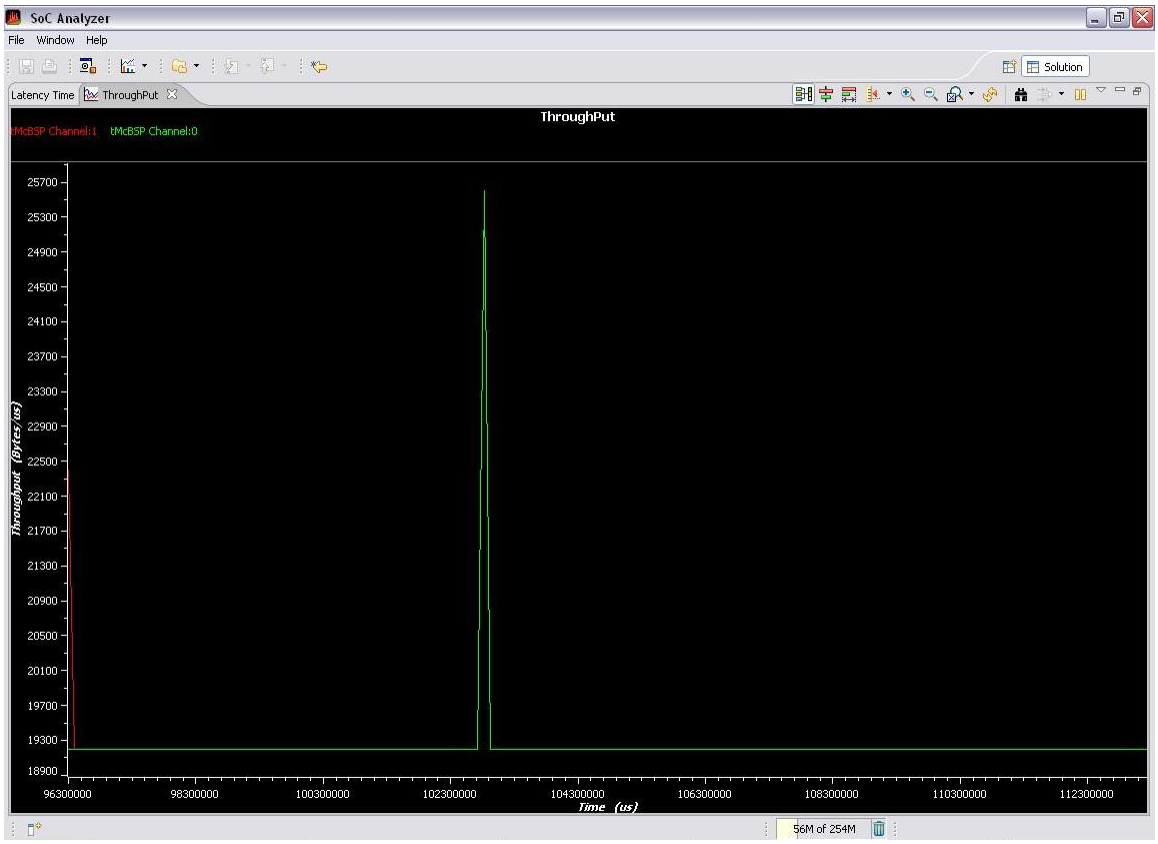
Performance and Benchmarks

MCBSP DEVICE DRIVER SUB-COMPONENT	PROGRAM MEMORY (IN BYTES)	DATA MEMORY (IN BYTES)		
		MEMORY TYPE		TOTAL
		INITIALIZED	UN INITIALIZED	
dda(iom)	2348	28	188	**2564**
ddc	18064	200	5204	**23468**
llc	4140	0	16	**4156**
Total	**24552**	**228**	**5408**	**30188**

- System Components Total Memory (Code & Data): **30188** Bytes



- Latency graph for McBSP receive and transmit channels



- Throughput graphs for McBSP receive and transmit channels

Note: The Driver Performance Characteristics can be included once testing is done on DM6437/C6424 SOC. The graphs are taken for McBSP operating in slave mode at 48 KHz sampling rate. For generating performance figures of McBSP driver in other modes please refer the top-level user guide for usage of SoC Analyzer.

All memory requirements are expressed in kilobytes (1 kilobyte = 1024 8-bit bytes) for **full-featured** device driver usage.
This data was gathered using the *.map file generated by the CCS.

Uninitialized data: .bss
Initialized data: .cinit + .const
Initialized code: .text + .text: init

References

[1] McBSP Module Hardware Specifications

[2] EDMA 3.0 Module Hardware Specifications

[3] SPRU943.pdf - TMS320DM643x DMP Multichannel Buffered Serial Port (McBSP) Interface User's Guide

Glossary

CSLR	TI Terminology, Chip Support Register Layer
IOM	TI Terminology, IO Mini layer
EDMA	TI Terminology, Enhanced Direct Memory Access Controller
LLC	TI Terminology, Lower Level Controller
DDC	TI Terminology, Device Driver Core