Supported Data Formats

Mcasp driver expects the data (samples) to be arranged in a specific format when requesting for an IO transfer. These formats are explained under scenario of using 1 serializer and 2 or more serializers. Some of the multi-channel DACs (such as WM8746) expects the samples for all the channels to be received over single serializers. To support these DACs, Mcasp provides support for couple of more data formats. The required buffer format could be configured at driver creation time. The sections below capture the details of supported data formats.

McASP Mode	Single Serializer	Multiple Serializer
Burst Mode / DSP Mode	Interleaved Data Format	Non-interleaved data format
TDM 1 Slot	Interleaved Data Format	Non-interleaved data format
Multi-Slots TDM	Interleaved Data Format Non-interleaveddata format	Non-interleaved data format Semi-interleaved data format
DIT	Interleaved Data Format	Non-interleaved data format

Interleave Data Format (Burst Mode / 1 Slot TDM mode / Multi-Slots TFM / DIT mode)

When configured as interleaved format, it is expected that McASP is configured to use 1 serializer. The expected data format is as depicted below.
[<**Slot1**-Sample**1**>, <**Slot1**-Sample**2**>…<**Slot1**-Sample**N**>]
The size (number of bytes) that would be required to specify during an IO request is computed using the formula size = <word width>*<number of samples **N**>. The sample application that came with this package demonstrates the use of this data format.

File audioSample_io.c implements the functions which configure McASP to use this buffer format.
The key configurations are

- `Mcasp_ChanParams.noOfChannels = 0x00`
- `Mcasp_ChanParams.noOfSerRequested = 0x01`
- `Mcasp_ChanParams.indexOfSersRequested[0] = SERIALIZER_0`
- The size of the IO request is computed as <`No of Bytes per Sample`> * <`No of Samples` >. This value should be given as a size parameter of SIO`_submit` ()
- Idle Time data pattern length computation. Minimum length should be **<word width in bytes>** or an integral multiple of computed value. While allocating buffer, allocate `<computed value> * <no of slots enabled>`.

Non-Interleaved Data Format (Burst Mode / 1 Slot TDM mode / Multi-Slots TDM / DIT mode)

When configured as non-interleaved format, it is expected that PSP driver is configured to use multiple serializers. The expected data format is as depicted below.
When configured to use multiple serializers, the samples are expected to be

contiguous for a serializer, as depicted below. The assumption here is no of serializers is 2 and no of samples is N
[<**Seriliazer1**-Sample**1**>, <**Seriliazer1**-Sample**2**>…<**Seriliazer1**-Sample**N**>,
<**Seriliazer2**-Sample**1**>, <**Seriliazer2**-Sample**2**>, <**Seriliazer2**-Sample**N**>,
<**Seriliazer3**-Sample**1**>, <**Seriliazer3**-Sample**2**>…<**Seriliazer3**-Sample**N**>]
The key configurations are

```
Mcasp_ChanParams.noOfChannels = 0x00
Mcasp_ChanParams.noOfSerRequested = 0x03
Mcasp_ChanParams.indexOfSersRequested[0] = SERIALIZER_0
Mcasp_ChanParams.indexOfSersRequested[1] = SERIALIZER_6
Mcasp_ChanParams.indexOfSersRequested[2] = SERIALIZER_8
```

- The size of the IO request is computed as <`No of Bytes per Sample`> * <`No of Samples` **`per Serializer`**>. This value should be given as a size parameter

- Idle Time data pattern length computation. Minimum length should be **`<word width in bytes>`** or an integral multiple of computed value. While allocating the buffer allocate `computed value * no of serializers enabled`.

Non-Interleaved Data Format (Multiple Slots Single serializer)

When configured to use multiple slots, one serializer and non-interleaved format. The samples are expected to be contiguous for a slot, as depicted below. The assumption here is no of slots is 2 and no of samples is N
[<**Slot1**-Sample**1**>, <**Slot1**-Sample**2**>…<**Slot1**-Sample**N**>,
<**Slot2**-Sample**1**>, <**Slot2**-Sample**2**>, <**Slot2**-Sample**N**>]
i.e. The samples of Slot1 are contiguous followed by contiguous samples of Slot 2
The key configurations are

- `Mcasp_ChanParams.noOfChannels = 0x00`
- `Mcasp_ChanParams.noOfSerRequested = 0x01`
- The size of the IO request is computed as <`No of Bytes per Sample`> * <`No of Samples` **`per Slot`**>. This value should be given as a size parameter

Idle Time[9.4] data pattern length computation. Minimum length should be **`<number of slots enabled> * <word width in bytes>`** or an integral multiple of computed value. While allocating the buffer, allocate <compute value> * <no of slots>
Consider as an example where the no of slots are 3 and no of samples per slot is N
[<**Slot1**-Sample**1**>, <**Slot1**-Sample**2**>…<**Slot1**-Sample**N**>,
<**Slot2**-Sample**1**>, <**Slot2**-Sample**2**>, <**Slot2**-Sample**N**>,
<**Slot3**-Sample**1**>, <**Slot3**-Sample**2**>…<**Slot3**-Sample**N**>]

Semi-Interleaved Data Format (Multiple Slots Multiple serializer)

When configured to use multi-slots with multi-serializer, the sample for all serializer for a give slot is contiguous, further the samples for all slots are interleaved. The following representation specifies the expected data format. The assumption in this example is we have enabled 2 serializer and two slots in each serializer.

[<**Slot1**-Sample**1**-Serializer**1**>, <**Slot1**-Sample**1**-Serializer**2**>,
<**Slot2**-Sample**2**-Serializer**1**>, <**Slot2**-Sample**2**-Serializer**2**>,…
<**Slot1**-Sample**N**-Serializer**1**>, <**Slot2**-Sample**N**-Serializer**2**>]

The key configurations are

- `Mcasp_ChanParams.noOfChannels = 0x00`
- `Mcasp_ChanParams.noOfSerRequested = 0x02`
- The size of the IO request is computed as <`No of Bytes per Sample`> * <`No of Samples` **`per Slot`**>. This value should be given as a size parameter.

Idle Time data pattern length computation. Minimum length should be **<number of slots enabled> * <word width in bytes>** or an integral multiple of computed value. While allocating memory for the loopJobBuffer allocate the `computed size * no of serializers enabled`