USB Audio

Revision 0.9.0-41360

May 6, 2015

List of Tables

List of Figures

NO TABLE OF FIGURES ENTRIES FOUND.

24 USB Audio

24.1 Overview

The USB Audio module handles streaming of audio over USB FS and HS interfaces compliant with the Class Driver Version 2.0.

Features
- USB Audio Class 2.0 Compliant
- Supports FS and HS modes
- Supports one full micro-frame per direction.
- Supports Asynchronous mode
- Supports Synchronous mode
- Supports configurable safety offsets

24.2 Typical programming sequence

[needs cleanup]

When the device is connected to a host the USB firmware will provide descriptors etc.

When the host selects a configuration with streaming enabled the firmware should:

- Write tx and rx cfg registers with the appropriate values (EN bits still 0).
- Clear the interrupts by writing all ones.
- If the mode is synchronous set the PLL to lock to SOH.
- Write tx and rx cfg registers with EN=1

After that the device can choose to use a polling or an interrupt driven scheme to monitor the status.

This is what can be expected:

Asynchronous mode:

The transmitter will likely indicate a single resync. Error and then no more errors.

It might generate incomplete errors as well if the host is slow to start asking for data.

The receiver will generate no data errors until it starts receiving data. It will also generate resync errors. After receiving data it will likely generate one or two resync errors as the host matches the outgoing packets to the incoming.

Synchronous mode:

The receiver will generate no data errors until it starts receiving data. It will also generate resync errors.

After receiving data it will generate resync errors as the PLL is locking. Please note that the PLL could be locked well ahead of time as soon as it is known that the mode is synchronous and what the sample rate is going to be.

The transmitter will generate incomplete errors until the host starts asking for data.

There might be a number of resync errors as the PLL is locking.

General:

A simple state machine can be made which is looking for a certain period of time with no errors. The system should be error free in less than 6 packets which is 750us in HS mode and 6ms in FS mode when running Asynchronous.

In synchronous mode it might take a little longer but the PLL can be trimmed to get a quick lock good enough for data integrity and then it can continue working on reducing the jitter.

24.3 Module Configuration

The USB Audio Control registers are addressed through 1 base address:

Table 24.1 USB Audio Control base address

Base address	Description
0xC9000000	DICE3_USB_AUD – USB Audio

Table 24.2 USB Audio register summary

Address Offset	Register	Description
0x0000	USB_CSR	*Global Control and Status Register*
0x0004	USB_RX_CFG0	*Rx Settings 0 Register*
0x0008	USB_RX_CFG1	*Rx Settings 1 Register*
0x000C	USB_TX_CFG0	*Tx Settings 0 Register*
0x0010	USB_TX_CFG1	*Tx Settings 1 Register*
0x0014	USB_IEN_SET	*Interrupt Enable Set Register*
0x0018	USB_IEN_CLR	*Interrupt Enable Clear Register*
0x001C	USB_IEN_STAT	*Raw Interrupt Status Register*

24.3.1 Control and Status Register – USB_CSR

Address offset: 0x0000 USB_CSR

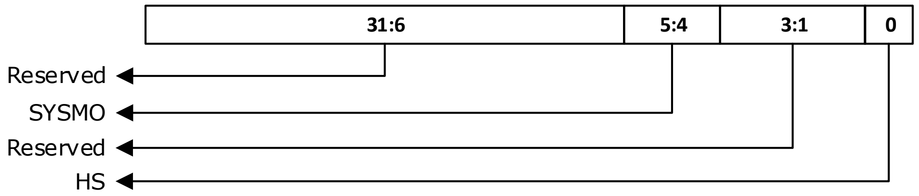

Table 24.3 Control and Status Register bit assignments

Name	Bit	Reset	Dir	Description
Reserved	31:6	0	N/A	Reserved
SYSMO	5:4	0	R/W	The sys_mode of the system (see note below): 00: low 01: mid 10: high
Reserved	3:1	0	N/A	Reserved
HS	0	0	R	Set if the USB is connected in HighSpeed mode else clear.

Note on SYSMO:

This parameter is only used to control how the FIFO allocates samples. It sets the compromise between channels and sample depth. In some cases where drivers with jittery software PLL's are used the SYSMO might have to be set higher to accommodate more safety. See sections *24.3.3.1* and *24.3.5.1* below for a discussion of the safety settings mentioned here.

Table 24.4 FIFO configurations

Speed	SYSMO	channels	depth
HS	0	64	16
HS	1	32	32
HS	2	16	64
FS	0	8	128
FS	1	4	256
FS	2	2	512

Please note that **SF_MAX** cannot be set higher than depth – **LEN_MAX** – 1.

SF_MAX < depth – LEN_MAX

For drivers with a large variation in data rate it might be necessary to use larger safety margins and the appropriate mode must be chosen.

Example: 44.1k HS with variation of 13 samples:

LEN_MIN = 5

LEN_MAX = 6

SFTY = 7

SF_MAX = 13

In this case SF_MAX + LEN_MAX = 19, so SYSMO = 1 should be used.

24.3.2 Rx Settings 0 Register – USB_ USB_RX_CFG0

Address offset: 0x0004 USB_ USB_RX_CFG0

This register, along with USB_ USB_RX_CFG1, is used for configuring the USB Receiver. Please note that changing these registers while the receiver is enabled (EN_RX is set) might lead to unpredictable behavior.

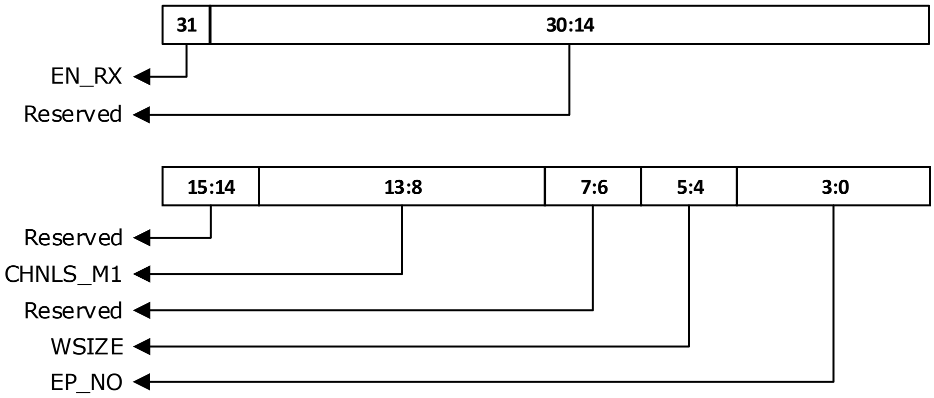

Table 24.5 Rx Settings 0 Register bit assignments

Name	Bit	Reset	Dir	Description
EN_RX	31	0	R/W	This bit enables the receiver. When set none of the other values should be changed.
Reserved	30:14	0	N/A	Reserved
CHNLS_M1	13:8	0	R/W	The number of channels less one. To receive 8 channels write 7.
Reserved	7:6	0	N/A	Reserved
WSIZE	5:4	0	R/W	The word size of the samples: 01: 16 bit 10: 24 bit 11: 32bit

Name	Bit	Reset	Dir	Description
EP_NO	3:0	0	R/W	The endpoint number to receive from.

24.3.3 Rx Settings 1 Register – USB_ RX_CFG1

Address offset: 0x0008 USB_ RX_CFG1

This register, along with USB_ USB_RX_CFG0, is used for configuring the USB Receiver. Please note that changing these registers while the receiver is enabled (EN_RX is set) might lead to unpredictable behavior.

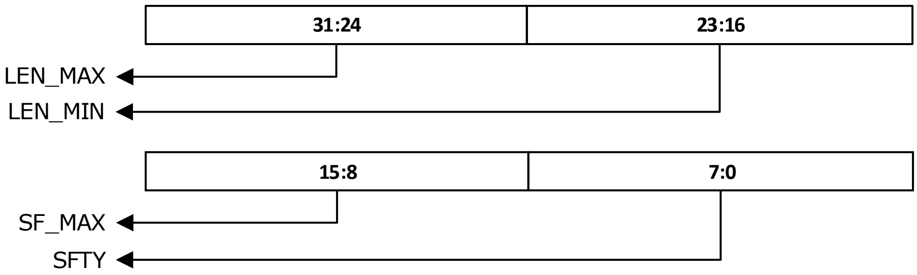

Table 24.6 Rx Settings 1 Register bit assignments

Name	Bit	Reset	Dir	Description
LEN_MAX	31:24	0	R/W	This field indicates the maximum number of samples which can be sent in one packet.
LEN_MIN	23:16	0	R/W	This field indicates the minimum number of samples which can be sent in one packet.
SF_MAX	15:8	0	R/W	The maximum safety offset in samples to accept before a resync. Is requested.
SFTY	7:0	0	R/W	The center safety offset.

24.3.3.1 Receiver lengths and safety settings

The USB Audio specifications require data to be packed in certain ways, depending on the mode.

In Asynchronous mode, data will be sent as they are available which depend on the audio clock and its relation to the USB frame interval.

If the clock for example is 48khz and it is in perfect sync with the USB frame, exactly 6 packets will be sent per frame. If it is asynchronous it can vary from 5 to 7 depending on the relation.

When the device is in Asynchronous mode the max and min length should be set as indicated in the table below. The same numbers can be used for Synchronous mode.

Table 24.7 Suggested Receiver min/max sample values

Rate	HS			FS		
	smpl/frm	min	max	smpl/frm	min	max
44.1k	5.512	4	6	44.1	43	45
48k	6	5	7	48	47	49
88.2k	11.025	10	12	88.2	87	89
96k	12	11	13	96	95	97
176.4k	22.05	21	23	176.4	175	177
192	24	23	25	192	191	193

The safety offset takes care of aligning the data to the sample clock. In both Asynchronous and Synchronous modes, the rate of incoming samples will be synchronous to the sample clock. In Asynchronous mode this is obtained by the host using the incoming pattern for outgoing packets. In Synchronous mode this is obtained by having both the host and the device synchronized to the USB frame.

As the size of the packet will vary it is important to always have enough data in the buffer even when a small packet arrives.

Initially, when the first packet arrives a resynchronize event will be generated which makes sure that the router takes samples **SFTY** ahead of the data that just arrived. As the router takes samples out and new packets come in the distance between the read and write pointers will fluctuate. This fluctuation should never be more than the maximum difference between a minimum and a maximum packet, but if the frequency is drifting and the host (in Asynchronous mode) is slow to repeat the patterns this could be more.

The default setting for **SFTY** should be 2 and for **SF_MAX** 4. This will result in a symmetrical safety around 2 with the ability to go up and down by 2 before the system will resynchronize.

For drivers with software PLL's with lots of jitter these numbers could be much higher. Choose the correct **SYSMO** to allow for the safety required.

The more lax the safety settings are, the less accurate the latency through the device from session to session will be.

24.3.4 Tx Settings 0 Register – USB_ TX_CFG0

Address offset: 0x000C USB_ TX_CFG0

This register, along with USB_ TX_CFG1, is used for configuring the USB Transmitter. Please note that changing these registers while EN_TX is set might lead to unpredictable behavior.

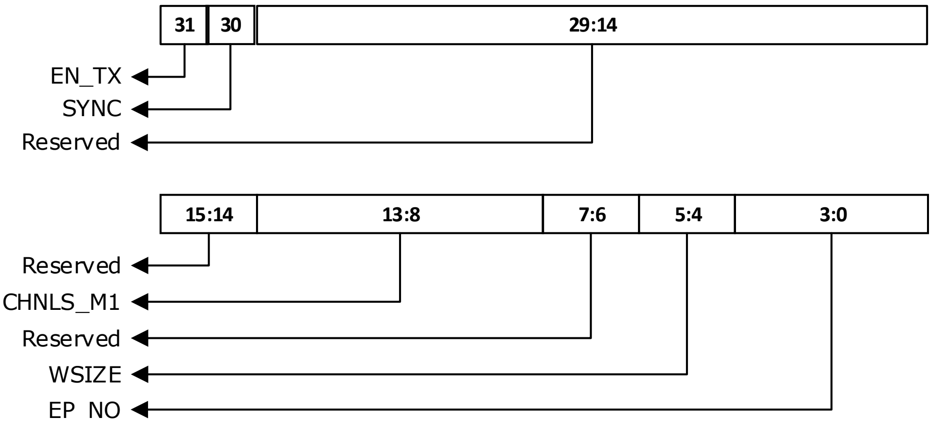

Table 24.8 Tx Settings 0 Register bit assignments

Name	Bit	Reset	Dir	Description
EN_RX	31	0	R/W	This bit enables the trensmitter. When set none of the other values should be changed.
SYNC	30	0	R/W	If set the transmitter is in synchronous mode and will use the last received packet length to set the transmitted packet length.
Reserved	29:14	0	N/A	Reserved
CHNLS_M1	13:8	0	R/W	The number of channels less one. To transmit 8 channels write 7.
Reserved	7:6	0	N/A	Reserved
WSIZE	5:4	0	R/W	The word size of the samples: 01: 16 bit 10: 24 bit 11: 32bit
EP_NO	3:0	0	R/W	The endpoint number to transmit to.

24.3.5 Tx Settings 1 Register – USB_ TX_CFG1

Address offset: 0x0010 USB_ TX_CFG1

This register, along with USB_ TX_CFG0, is used for configuring the USB Transmitter. Please note that changing these registers while EN_TX is set might lead to unpredictable behavior.

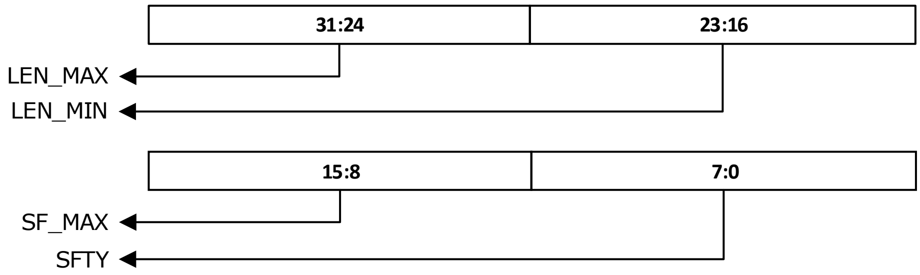

Table 24.9 Tx Settings 1 Register bit assignments

Name	Bit	Reset	Dir	Description
LEN_MAX	31:24	0	R/W	This field indicates the maximum number of samples which can be sent in one packet.
LEN_MIN	23:16	0	R/W	This field indicates the minimum number of samples which can be sent in one packet.
SF_MAX	15:8	0	R/W	The maximum safety offset in samples to accept before a resync is requested.
SFTY	7:0	0	R/W	The center safety offset.

24.3.5.1 Transmitter lengths and safety settings

The USB Audio specifications require data to be packed in certain ways, depending on the mode.

In Asynchronous mode, data will be sent as they are available which depend on the audio clock and its relation to the USB frame interval.

If the clock for example is 48khz and it is in perfect sync with the USB frame, exactly 6 packets will be sent per frame. If it is asynchronous it can vary from 5 to 7 depending on the relation.

When the device is in Asynchronous mode the max and min length should be set as indicated in the table below.

Table 24.10 Suggested Transmitter min/max values, Asynchronous mode

Rate	HS			FS		
	smpl/frm	min	max	smpl/frm	min	max
44.1k	5.512	5	6	44.1	44	45
48k	6	5	7	48	47	49
88.2k	11.025	11	12	88.2	88	89
96k	12	11	13	96	95	97
176.4k	22.05	22	23	176.4	176	177
192	24	23	25	192	191	193

And for Synchronous mode:

Table 24.11 Suggested Transmitter min/max values, Synchronous mode

Rate	HS			FS		
	smpl/frm	min	max	smpl/frm	min	max
44.1k	5.512	5	6	44.1	44	45
48k	6	6	6	48	48	48
88.2k	11.025	11	12	88.2	88	89

Rate	HS			FS		
	smpl/frm	min	max	smpl/frm	min	max
96k	12	12	12	96	96	96
176.4k	22.05	22	23	176.4	176	177
192	24	24	24	192	192	192

The safety offset takes care of aligning the data to the sample clock.

For the transmitter this differs between Asynchronous and Synchronous modes. In Asynchronous mode the flow is controlled from the device and the safety should only ever move by +/-1.

In Synchronous mode the device must sent data by mirroring the packets coming in on the receiver. Jitter between the USB clocks and the audio clock (which are synchronized) can result in variations in the amount of data in the FIFO.

Initially, when the first packet is requested a resynchronize event will be generated which makes sure that the transmitter takes samples **SFTY** ahead of route write pointer. As the router puts samples in and new packets are transmitted the distance between the read and write heads will fluctuate. This fluctuation should never be more than the maximum difference between a minimum and a maximum packet but if the frequency is drifting and the PLL (in Sync. mode) is slow to track the change this could be more.

It is suggested that the safety parameters are set as follows:

Table 24.12 Transmitter safety values

Param	Asynchronous	Synchronous
SFTY	1	2
SF_MAX	2	4

The more lax the safety settings are, the less accurate the latency through the device from session to session will be.

24.3.6 Interrupt Enable Set Register – USB_ IEN_SET

Address offset: 0x0014 USB_ IEN_SET

This register is used to enable the USB Audio related interrupts. The current status can always be read from either USB_ IEN_SET or USB_ IEN_CLR. In order to set bits use the USB_ IEN_SET register with the bits set which should be set.

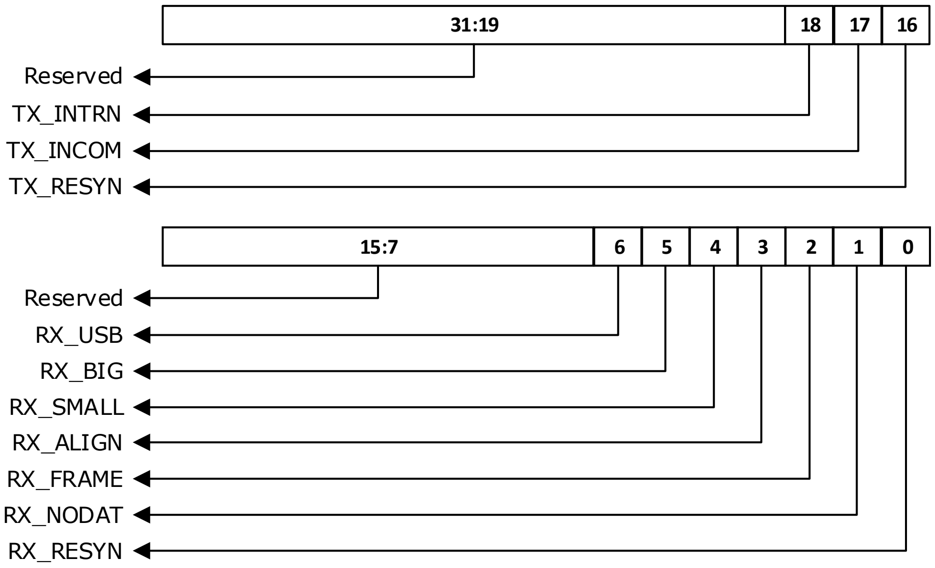

Table 24.13 Interrupt Enable Set Register bit assignments

Name	Bit	Reset	Dir	Description
Reserved	31:19	0	N/A	Reserved
TX_INTRN	18	0	R/W	Internal error, this should never be set.
TX_INCOM	17	0	R/W	Indicates that a frame interval has gone by without an IN token received. This will typically happen when the host stops the device and the transmitter has not yet been stopped (disabled). If it happens during normal operation it is likely a host problem or an IN token was corrupted.
TX_RESYN	16	0	R/W	Indicates that resynchronization was done. The transmitter either had too much data accumulated or ran out of data. This is guarded by the safety parameters.
Reserved	15:7	0	N/A	Reserved
RX_USB	6	0	R/W	Indicates that a packet was received with a USB error. This is typically an indication of a hardware error.
RX_BIG	5	0	R/W	Indicates that a packet was too big. This is typically because the settings of the receiver do not match the settings of the host.
RX_SMALL	4	0	R/W	Indicates that a packet was too small. This is typically because the settings of the receiver do not match the settings of the host.
RX_ALIGN	3	0	R/W	Indicates that the packet was not aligned to an integer number of samples. This is typically because the settings of the receiver do not match the settings of the host.
RX_FRAME	2	0	R/W	Indicates that a new frame has started before receiving the previous packet was complete. This is probably never going to happen.

Name	Bit	Reset	Dir	Description
RX_NODAT	1	0	R/W	Indicates that a frame interval has gone by without an OUT token and data received. This will typically happen when the host stops the device and the transmitter has not yet been stopped (disabled). If it happens during normal operation it is likely a host problem or an IN token was corrupted.
RX_RESYN	0	0	R/W	Indicates that resynchronization was done. The receiver either had too much data accumulated or ran out of data. This is guarded by the safety parameters.

24.3.7 Interrupt Enable Clear Register – USB_ IEN_CLR

Address offset: 0x0018 USB_ IEN_CLR

This register is used to disable the USB Audio related interrupts. The current status can always be read from either USB_ IEN_SET or USB_ IEN_CLR. In order to clear bits use the USB_ IEN_CLR register with the bits set which should be cleared.

The register layout is the same as USB_ IEN_SET above.

24.3.8 Raw Interrupt Status Register – USB_ IEN_STAT

Address offset: 0x001C USB_ IEN_STAT

This register shows the current interrupt status (raw). All the bits are sticky and can be cleared by writing a ‘1’ to the corresponding bit.

The register layout is the same as USB_ IEN_SET above.

24.4 Revisions

Table 24.14 Document revision history

Date	Rev.	By	Change
May 6, 2015	0.9.0-41360	BK	Initial publication