AN13629

How to Switch I2S Bit Clock Coherently on Kinetis L Series MCU

Rev. 0 — 13 May 2022 Application Note

1 Introduction

Most parts of the Kinetis L series MCU integrate the same I2S (SAI) module, but the function of I2S module in different parts is not the same. For example, the FIFO size of the KL28 I2S module is 4x32 bits, and the FIFO size of other parts is 1x32 bits, among which KL0x does not have the I2S module. The parts with the I2S module in Kinetis L series MCU are shown in Table 1.

Contents

1 Introduction....1
2 How to switch the BCLK of the I2S module....3
3 Test....11
4 Conclusion....11
5 References....12
6 Revision history....12
Legal information....13

Table 1. The parts with I2S module in Kinetis L series MCU

Kinetis L series	Part number	Flash	RAM (kB)	I2S number	I2S FIFO size	USB	PLL	DMA channel	MCLK fractional divider	Supports USB audio application
KL1x	KL16Z32	32	4	1	1	0	YES	4	YES	NO
	KL16Z64	64	8	1	1	0	YES	4	YES	NO
	KL16Z128	128	16	1	1	0	YES	4	YES	NO
	KL16Z256	256	32	1	1	0	YES	4	YES	NO
	KL17Z128	128	16	1	1	0	NO	4	NO	NO
	KL17Z256	256	32	1	1	0	NO	4	NO	NO
KL2x	KL26Z128	128	16	1	1	1	YES	4	YES	YES
	KL26Z256	256	32	1	1	1	YES	4	YES	YES
	KL26Z32	32	4	1	1	1	YES	4	YES	YES
	KL26Z64	64	16	1	1	1	YES	4	YES	YES
	KL27Z128	128	16	1	1	1	NO	4	NO	YES, only asynchronous mode
	KL27Z256	256	32	1	1	1	NO	4	NO	YES, only asynchronous mode

Table continues on the next page...

Table 1. The parts with I2S module in Kinetis L series MCU (continued)

Kinetis L series	Part number	Flash	RAM (kB)	I2S number	I2S FIFO size	USB	PLL	DMA channel	MCLK fractional divider	Supports USB audio application
	KL28Z512	512	128	1	4	1	YES	8	NO	YES, only asynchronous mode
	KL33Z128	128	16	1	1	0	NO	4	NO	NO
	KL33Z256	256	32	1	1	0	NO	4	NO	NO
	KL36Z128	128	16	1	1	0	YES	4	YES	NO
KL3x	KL36Z256	256	32	1	1	0	YES	4	YES	NO
	KL43Z128	128	16	1	1	1	NO	4	NO	YES, only asynchronous mode
	KL43Z256	256	32	1	1	1	NO	4	NO	YES, only asynchronous mode
	KL46Z128	128	16	1	1	1	YES	4	YES	YES
KL4X	KL46Z256	256	32	1	1	1	YES	4	YES	YES

Table 1 lists that:

1. There are some parts with audio master clock (MCLK) fraction dividers, such as KL26 and KL46.
2. There are some parts without MCLK fractional dividers, such as KL27Z256, KL43Z256, and so on.
3. The FIFO size of the KL28 I2S transceiver is 4x32 bits, and FIFO size of the I2S module on other parts is 1x32 bits.
4. KL1x and KL3x do not have USB peripherals, while KL2x and KL4x have USB peripherals, and are suitable for USB audio applications.

NOTE

The parts without MCLK fractional divider can also be applied to USB audio application, such as USB audio devices working in asynchronous mode.

The device in asynchronous mode does not need to adjust the I2S clock on the device side, but only must tell the USB host the real sample rate of the I2S through the feedback endpoint. The USB host adjusts the transmission rate of the audio data. The MCU with MCLK fractional divider can be perfectly applied to USB audio applications, that is, it can realize the USB audio device in asynchronous mode and the USB audio device in synchronous mode.

There is a special use case to implement a synchronous mode USB audio device using an MCU without a fractional divider, as described in *USB Audio Synchronous Mode on KL27* (document AN13364). Figure 1 shows the USB audio synchronous mode is successfully implemented on KL27.

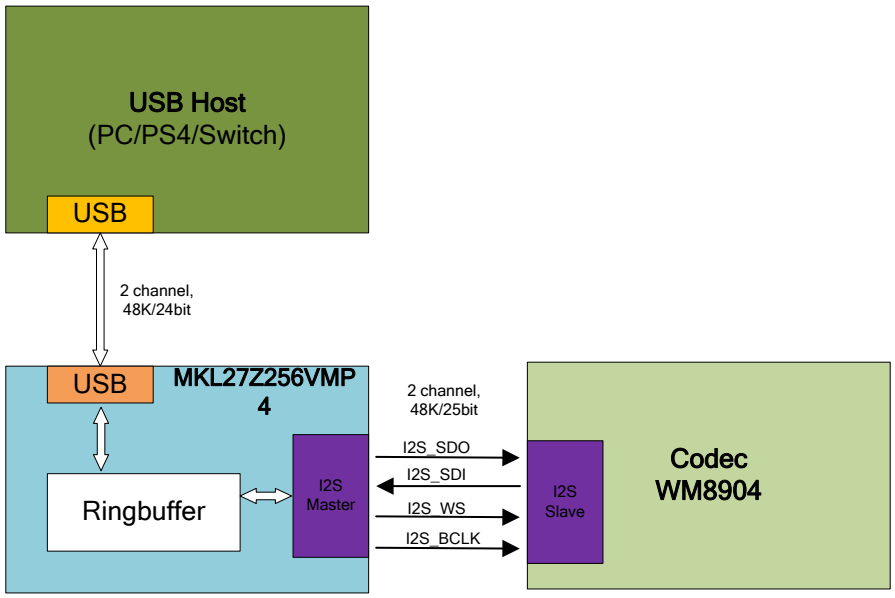

Figure 1. The system block diagram of KL27 audio speaker

The implementation method is to adjust the bit clock (BCLK) frequency by dynamically adjusting the integer frequency division factor and word length of the I2S transmitter (TX). Using this method, the I2S clock on the USB device side synchronizes with the start of frame (SOF) signal sent by the USB host. This application note introduces how to switch the bit clock of the I2S module coherently when the I2S module is working.

In the application of the USB audio synchronization mode, it is sometimes necessary to switch the BCLK of the I2S in real time to keep the I2S clock on the device side synchronized with the SOF signal sent by the USB host. DMA is used to implement the transfer of audio data between buffers and the I2S data registers due to the large amount of audio data and high real-time requirements. For example, in a USB audio speaker application, use DMA to transfer the audio data sent by the USB host to the I2S TX DATA register (TDR). Due to the high requirements for real-time performance, I2S data must be continuously transmitted, so the DMA link method is used to realize the continuous transmission of DMA. This application note takes KL27 as an example to describe how to switch the BCLK coherently when the DMA is working continuously.

2 How to switch the BCLK of the I2S module

This section uses the use case from *USB Audio Synchronous Mode on KL27* (document AN13364) as an example to explain how to switch the BCLK of the I2S module.

2.1 Introduction to KL27 I2S module

The block diagram of the I2S module of the Kinetis L series MCU is shown in Figure 2.

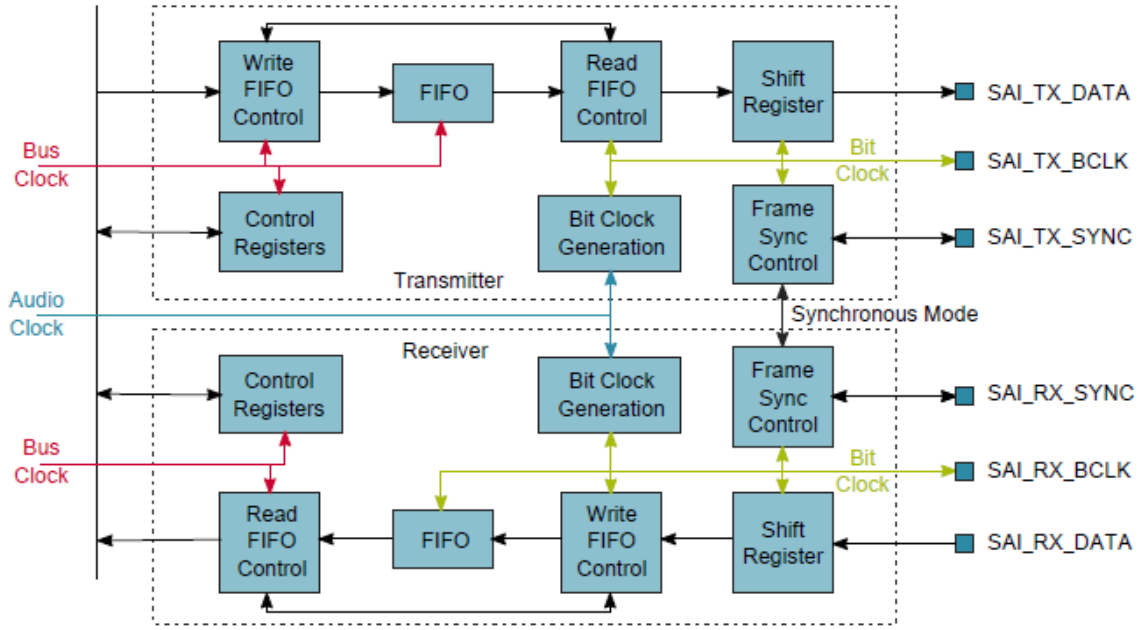

Figure 2. I2S/SAI block diagram

The audio clock in Figure 2 is used to generate the bit clock when the receiver or transmitter is configured for an internally generated bit clock. Each I2S (SAI) peripheral can control the input clock selection, pin direction, and divide ratio of one audio master clock. If an I2S module using the audio master clock is enabled, the input clock selection and pin direction cannot be altered. The MCLK divide ratio can be altered while an I2S is using the master clock, although the change in the divide ratio takes several cycles. The audio master clock generation and selection is chip-specific. For more information on how the audio master clocks are generated, see chip-specific reference manual *KL27 Sub-Family Reference Manual* (document KL27P64M48SF6RM). A typical implementation appears in the Figure 3.

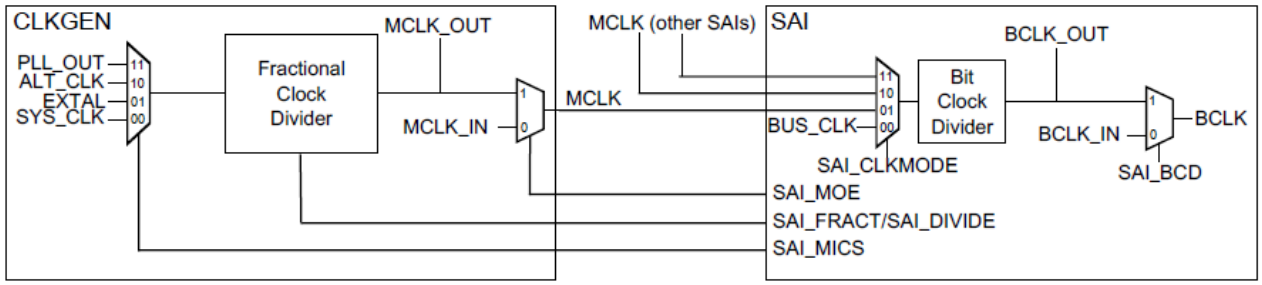

Figure 3. I2S master clock generation

NOTE

Figure 3 is only a typical diagram, the input selection of the master clock of different chips is different, and not all chips have fractional clock dividers. The real I2S clock block diagram of KL27 is shown in Figure 4.

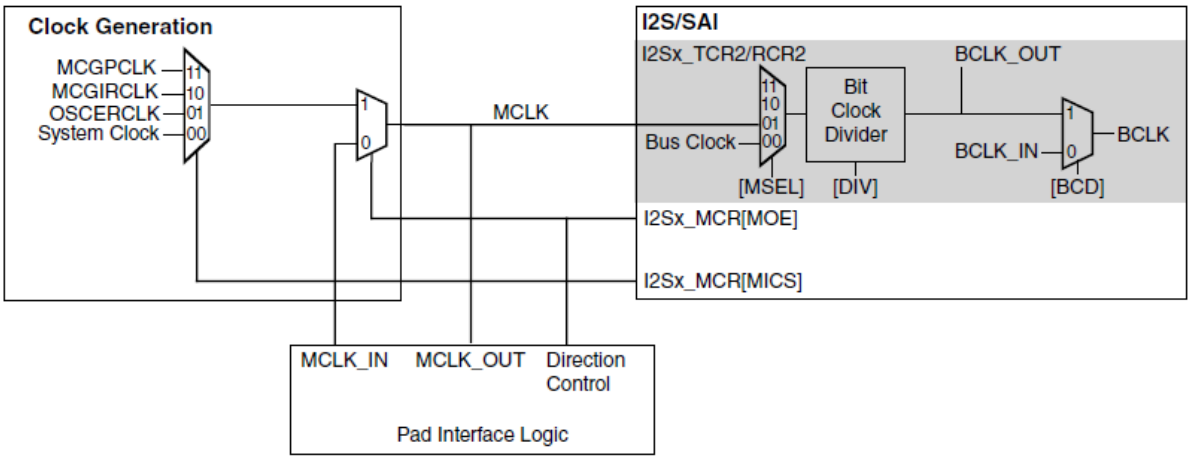

Figure 4. KL27 I2S clock generation

Figure 4 shows that there are four options for the clock source of the master clock of KL27. Table 2 lists the KL27 I2S MCLK input clock selection.

Table 2. KL27 I2S MCLK input clock selection

MCR[MICS]	Clock selection
00	System clock
01	OSCERCLK
10	MCGIRCLK
11	MCGPCLK

In *USB Audio Synchronous Mode on KL27* (document AN13364), the input source of I2S MCLK is 48 M system clock. KL27 does not have PLL and master clock divider (fractional divider). So if you want to change the BCLK of the I2S module, you can only modify the bit clock divider for a given bit clock source. The transmitter and receiver can independently select between the bus clock and the audio master clock (MCLK) to generate the bit clock. The MCLK select (MSEL) field of the transmit configuration 2 register and receive configuration 2 register (TCR2[MSEL] and RCR2[MSEL]) selects the master clock.

Table 3 shows the TCR2[MSEL] and RCR2[MSEL] field settings for KL27.

Table 3. KL27 I2S master clock settings

TCR2[MSEL], RCR2[MSEL]	Master clock
00	Bus clock
01	I2S0_MCLK
10	Not supported
11	Not supported

In *USB Audio Synchronous Mode on KL27* (document AN13364), I2S_MCLK is used as the clock source of BCLK.

2.2 Update BCLK

As mentioned in the Introduction to KL27 I2S module, KL27 does not have a PLL and a fractional divider. Therefore, a bit clock that is closer to the ideal value is only obtained by modifying the bit clock divider (I2S_TCR2[DIV]) or I2S word length (I2S_TCR4[SYWD]).

This section describes how to choose the correct moment to modify the bit clock divider and I2S word length so that the bit clock can be switched coherently. According to the description of the *KL27 Sub-Family Reference Manual* (document KL27P64M48SF6RM), when the I2S_TCSR[TE] bit is set, the I2S_TCR2, I2S_TCR4, I2S_TCR5 registers cannot be changed. Among them, I2S_TCR2[DIV] is used to configure the frequency division coefficient of the bit clock, and I2S_TCR4[SYWD] is used to configure the word length of I2S. Therefore, the I2S_TCSR[TE] bit must be cleared before modifying these registers.

To sum up, modifying the frequency of the bit clock on the KL27 requires the following three steps:

1. Clear I2S_TCSR[TE] bit
2. Modify bit clock divider (I2S_TCR2[DIV]) and I2S word length (I2S_TCR4[SYWD])
3. Re-enable I2S_TCSR[TE] bit

In *USB Audio Synchronous Mode on KL27* (document AN13364), since the DMA link method is used, the DMA is always working. When updating the BCLK, it is necessary to select a suitable time so that the BCLK can be updated coherently without TX FIFO underrun.

How to properly clear TE describes how to find this correct moment to clear TE. The TE is not cleared until the transmission of the current frame is completed. After the transmission of the current frame is completed, the frame sync/word select (WS) signal is idle. In *USB Audio Synchronous Mode on KL27* (document AN13364), the idle state of WS is high.

Before introducing how to update BCLK correctly, let us first introduce the initial configuration of the I2S registers used in this application note.

2.2.1 KL27 I2S register configuration

The format of the USB audio data used in this example is two-channel 48 K/24 bit. DMA transfers the audio data in the ring buffer to the I2S TX DATA register, and the DMA transfer does not support the 24-bit format. Thus, the data width of each transfer of the DMA is configured as 32 bits. In order to facilitate the DMA transfer, after receiving the 24-bit audio data sent by the USB host, fill the upper 8 bits with 0. Then save it in the ring buffer and wait for the DMA transfer to the I2S TX DATA register. Therefore, the data format of I2S TX channel should be configured as shown in Table 4.

Table 4. KL27 I2S TX format

I2S TX format		Description
Frame size	2	Two channel
Sync width	32	I2S data width
Sample rate	48000	Sample rate

The required BCLK = 48000 * 2 * 32 = 3072000. This frequency cannot be obtained by 48 M MCLK through integer frequency division, but we can change the I2S word length from 32 to 25. For details, refer to *Chapter 2.4 KL27 clock distribution* in *USB Audio Synchronous Mode on KL27* (document AN13364).

The value of I2S_TCR2[DIV] = 9, and the frequency division coefficient = (I2S_TCR2[DIV] + 1) * 2 = 20.

BCLK = 48 M / 48000 / 2 /25 / ((DIV+1) * 2) = 2.4 M.

In this application note, the configuration of related registers of I2S TX is shown in Table 5.

Table 5. KL27 I2S TX related register configuration

Register	Field	Value	Description
I2S_TCSR	FWDE	1	Enables the DMA request
	FEIE	1	Enables the interrupt
	BCE	1	Transmit bit clock is enabled
	TE	1	Transmitter is enabled. When software clears this field, the transmitter remains enabled, and this bit remains set, until the end of the current frame.
I2S_TCR2	DIV	9	BCLK frequency division factor = (9 + 1) * 2
	BCD	1	Bit clock is generated internally in Master mode.
	BCP	1	Bit clock is active low with drive outputs on the falling edge and sample inputs on the rising edge.
	SYNC	0	Asynchronous mode
I2S_TCR4	FSD	1	Frame sync is generated internally in Central mode.
	FSP	1	Frame sync is active low.
	ONDEM	0	Internal frame sync is generated continuously.
	FSE	1	Frame sync asserts 1 bit before the first bit of the frame.
	MF	1	MSB is transmitted first.
	SYWD	0x18	Word length = SYWD + 1
	FCONT	1	On FIFO error, the I2S will continue from the same word that caused the FIFO error to set after the FIFO warning flag has been cleared.
	FRSZ	1	Frame size is 2 words.
I2S_TCR5	W0W	0x18	Word 0 width is 25
	WNW	0x18	Word N width is 25.
I2S_MCR	DUF	0	MCLK divider ratio is not being updated currently.
	MOE	1	MCLK signal pin is configured as an output from the MCLK divider.
	MICS	0	System clock selected.

The input source of I2S_MCLK is the system clock of 48 M, and this clock generates BCLK.

From the configuration of I2S_TCR4 and I2S_TCR5, it can be seen that each frame contains 2 words, that is, two channels. The width of each word is 25 bits. The WS is active low and arrives one clock earlier than BCLK. The I2S signal is shown in Figure 5.

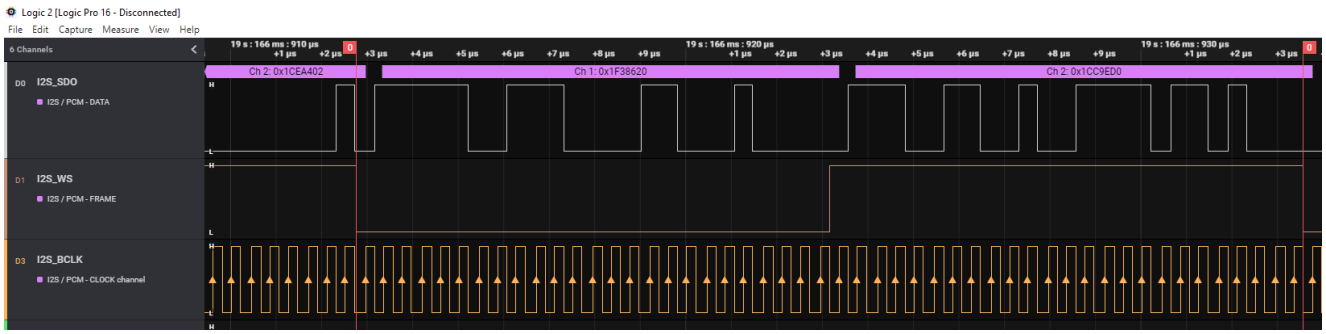

Figure 5. I2S signal

When software clears I2S_TCSR[TE], TE is not cleared immediately. It is cleared after the current frame transmission is completed, that is, TE can be cleared only when WS is high.

When TE is cleared, modify the bit clock divide I2Sx_TCR2[DIV] and word length I2Sx_TCR4[SYWD]. Then re-enable TE, the new bit clock, and WS takes effect immediately.

2.2.2 How to properly clear TE

The section describes that TE is cleared when WS is high. Thus, when DMA continues to work, find the appropriate moment as TE cannot be cleared at any time. If TE is cleared at any time, sometimes the operation of clearing TE conflicts with the falling edge of the next WS (start of the next frame). As a result, TE is cleared when WS is low (a new frame transmission starts), as shown in Figure 6.

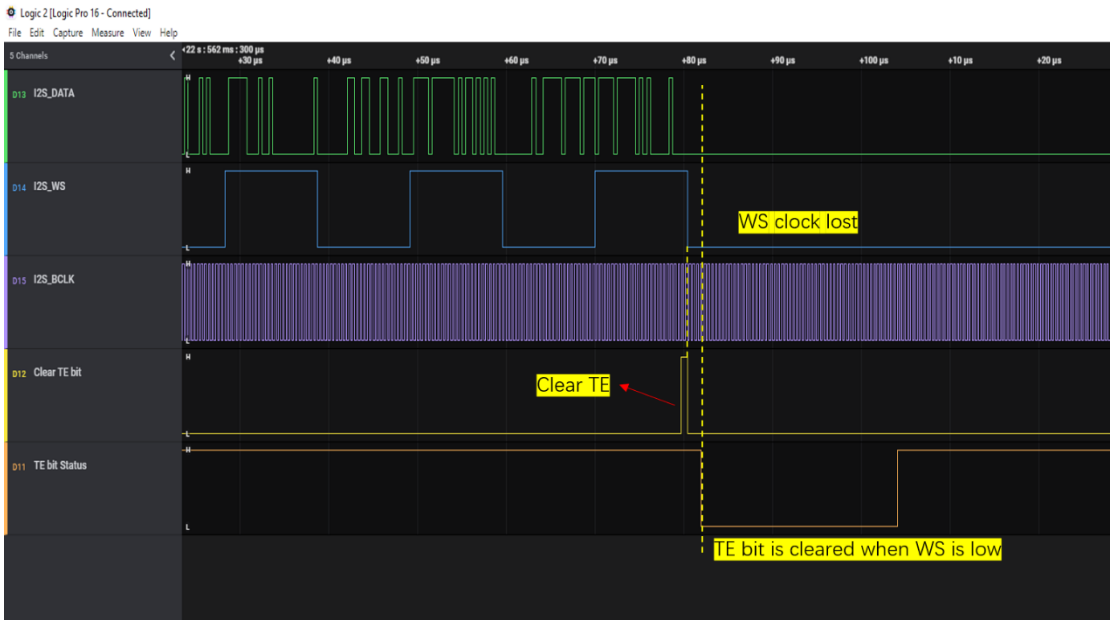

Figure 6. Clear TE when WS is low

After the bit clock divider and I2S word length are updated and TE is re-enabled, the new WS signal does not appear. Figure 6 shows the operation of clearing the TE bit (I2S0->TCSR = ((I2S0->TCSR & 0xFFE3FFFFU) & (~I2S_TCSR_TE_MASK))) is executed when the current frame is about to end. As a result, the operation of clearing TE conflicts with the start time (falling edge of WS) of the next frame. This phenomenon does not occur every time the BCLK is updated. It only occurs when there is a timing conflict between the operation of clearing TE and the falling edge of WS. The normal sequence of updating the bit clock may be shown in Figure 7 and Figure 8.

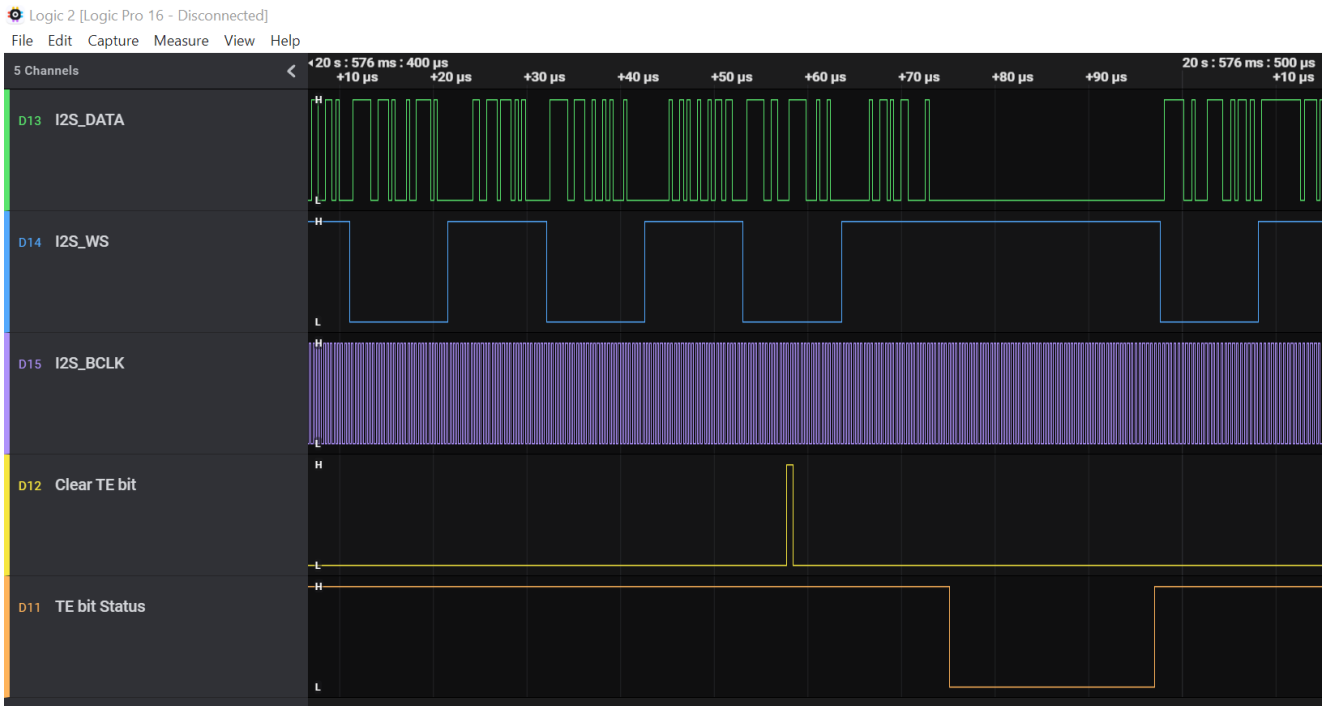

Figure 7. Clear TE in left channel stage

Figure 8. Clear TE in right channel stage

The operation of clearing TE may occur in the left channel stage or in the right channel stage, and there is no time conflict with the falling edge of WS, as shown in Figure 7 and Figure 8. In this case, TE is cleared after the current frame transmission ends, that is, when WS is high. Because the operation of clearing TE can occur at any time, there may be some potential risks as shown in Figure 6. Therefore, it is necessary to find an appropriate time to clear the TE bit to ensure that after each update of BCLK, the WS signal can work normally.

To make users better understand some behaviors inside the I2S module, several key time points are introduced.

- Moment when the data in the TX FIFO is loaded into the shift register:

  When the rising or falling edge of WS occurs, the data in the TX FIFO is loaded into the shift register. The KL27 has only one FIFO, so when writing data to the I2S TX DATA register, the data is written directly to the TX FIFO.

- Moment when FIFO empty flag is set:

  Once the data in the FIFO is loaded into the shift register and the FIFO is empty, the FIFO empty flag is set. DMA request is initiated to request the DMA to transfer new data to the TX I2S register. For example, when the falling edge of WS comes, the left channel data in the FIFO is loaded into the shift register, and the FIFO becomes empty. A new DMA request is initiated, and the DMA immediately writes the right channel data to the TX DATA register. This also means that the data of the right channel has been written into the FIFO when the shifter has not completed the transmission of the data of the left channel.

- Moment when FIFO error flag is set:

  If the corresponding bit of TCR3[TCE] is clear or if the FIFO is full, writes to a TDR are ignored. If the Transmit FIFO is empty, the TDR must be written at least three-bit clocks before the start of the next unmasked word to avoid a FIFO underrun. If the DMA stops working for some reason (for example, the I2S DMA request is disabled) and the I2S TX FIFO is empty, if there is a new WS rising or falling edge, the shift register cannot be loaded with new data, the FIFO error flag sets. If the FIFO error interrupt is enabled, a FIFO error interrupt also generates. Therefore, when clearing TE, not only ensure that TE is cleared when WS is high, but also ensure that new data has been written in the FIFO. If no new data is written into the FIFO, a FIFO error interrupt generates when the new bit clock takes effect, and the new WS falling edge appears because the TX FIFO is underrun.

The operation of clearing TE should be kept as far away as possible from the moment when the falling edge of WS occurs to prevent the time conflict between the operation of clearing TE and the falling edge of WS. The following methods can be used to find an appropriate time:

```
/* Wait TX FIFO is empty */
while(!(I2S0->TCSR & I2S_TCSR_FWF_MASK)) { }
```

Use the while loop in the program to poll the I2S0_TCSR register until the TX FIFO is empty. It means that the rising or falling edge of WS has arrived, the data in the FIFO has been loaded into the TX shift register, and a new DMA request has been initiated (DMA transfers new data in the FIFO to eliminate the FIFO empty flag). At this time, there is still about one word or two words before the falling edge of the next WS (start of the next frame), and the operation of clearing TE can be executed, as shown below:

```
/* Clear TE bit */
I2S0->TCSR = ((I2S0->TCSR & 0xFFE3FFFFU) & (~(I2S_TCSR_TE_MASK)));
```

After executing the above code, do not update the bit clock divider and I2S TX word length immediately because TE is not cleared until the current frame is transmitted. Therefore, it is necessary to wait for TE to be cleared, and use the following code to wait for TE to be cleared:

```
/* Wait until TE is cleared */
while ((I2S0->TCSR & I2S_TCSR_TE_MASK)) {}
```

Once TE becomes 0, the bit clock divider and I2S TX word length can be modified.

```
/* Modify the frequency division coefficient of the bit clock and the I2S word length*/
I2S0->TCR5 &= ~I2S_TCR5_WNW_MASK;
I2S0->TCR5 &= ~I2S_TCR5_WNW_MASK;
I2S0->TCR5 &= ~I2S_TCR5_W0W_MASK;
I2S0->TCR5 |= I2S_TCR5_WNW(wordLength - 1U) | I2S_TCR5_W0W(wordLength - 1U);
I2S0->TCR4 &= ~I2S_TCR4_SYWD_MASK;
I2S0->TCR4 |= I2S_TCR4_SYWD(wordLength-1);
I2S0->TCR2 |= I2S_TCR2_DIV(X);
```

After modifying the bit clock divider and TX word length, re-enable TE to make the new bit clock and WS take effect.

```
/* Enable TE */
SAI_TxEnable(AUDIO_SERVICE_I2S_BASEADDR, true);
```

The timing diagram of this process is shown in Figure 9.

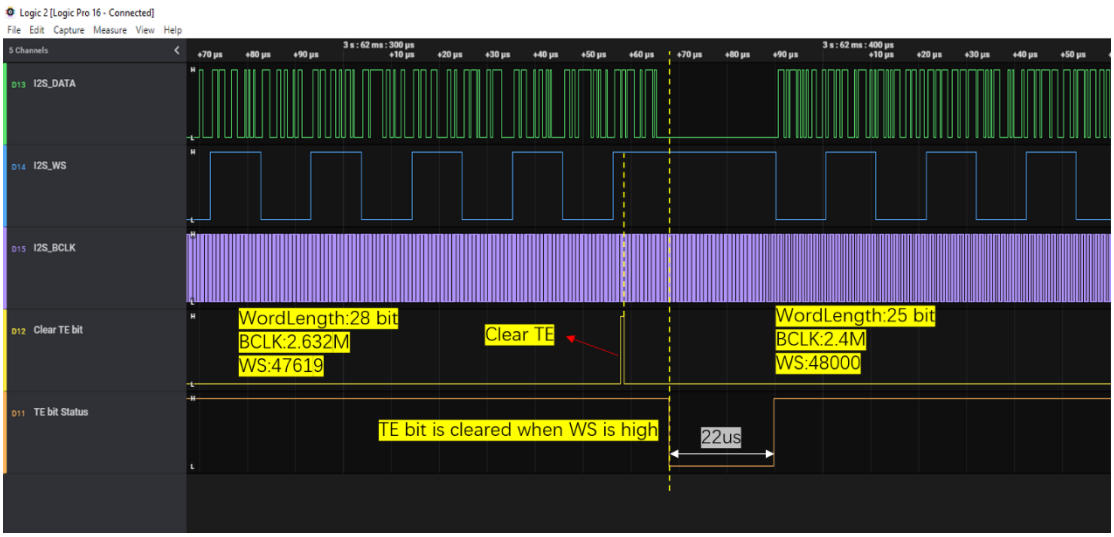

Figure 9. The correct timing for updating bit clock

As can be seen from Figure 9, the whole process of switching the bit clock takes 22 µs, and there is no data loss. It can be said that the bit clock is coherently updated. For the specific implementation code, refer to *USB Audio Synchronous Mode on KL27* (document AN13364) software.

3 Test

For the test environment and test details, refer to *USB Audio Synchronous Mode on KL27* (document AN13364). Table 6 lists the following three configurations that are selected for *USB Audio Synchronous Mode on KL27* (document AN13364).

Table 6. Configuration of three I2S_WS frequencies

I2S_WS	Word length	I2S_TCR2[DIV]	g_I2sFormatIndex
47619	28	8	0
48000	25	9	1
48387	31	7	2

The process of updating the BCLK is shown in Figure 9. The BCLK is switched from 2.632 M to 2.4 M, the WS frequency is switched from 47619 Hz to 48000 Hz, and the whole process only takes 22 µs.

4 Conclusion

This application note describes how to update the I2S bit clock coherently and WS frequency in USB audio applications without PLL and fractional dividers. The switching process of bit clock and WS only takes about 22 µs, which can be successfully implemented even when DMA is working continuously. For those MCUs with PLL and fractional divider, the MCLK divide ratio (I2S_MDR[FRACT]) can be directly modified to obtain the desired frequency. This process may take several clocks.

5 References

- *USB Audio Synchronous Mode on KL27* (document AN13364)
- *KL27 Sub-Family Reference Manual* (document KL27P64M48SF6RM)
- *KL26 Sub-Family Reference Manual with Addendum* (document KL26P121M48SF4RM)
- *KL46 Sub-Family Reference Manual* (document KL46P121M48SF4RM)

6 Revision history

Table 7 summarizes the changes done to this document since the initial release.

Table 7. Revision History

Revision number	Date	Substantive changes
0	13 May 2022	Initial release

Legal information

Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors and its suppliers accept no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this data sheet expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer's applications and products. Customer's responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer's applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately.

Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP.

NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

Trademarks

Notice: All referenced brands, product names, service names, and trademarks are the property of their respective owners.

NXP — wordmark and logo are trademarks of NXP B.V.

AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile — are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved.

Airfast — is a trademark of NXP B.V.

Bluetooth — the Bluetooth wordmark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by NXP Semiconductors is under license.

Cadence — the Cadence logo, and the other Cadence marks found at www.cadence.com/go/trademarks are trademarks or registered trademarks of Cadence Design Systems, Inc. All rights reserved worldwide.

CodeWarrior — is a trademark of NXP B.V.

ColdFire — is a trademark of NXP B.V.

ColdFire+ — is a trademark of NXP B.V.

EdgeLock — is a trademark of NXP B.V.

EdgeScale — is a trademark of NXP B.V.

EdgeVerse — is a trademark of NXP B.V.

eIQ — is a trademark of NXP B.V.

FeliCa — is a trademark of Sony Corporation.

Freescale — is a trademark of NXP B.V.

HITAG — is a trademark of NXP B.V.

ICODE and I-CODE — are trademarks of NXP B.V.

Immersiv3D — is a trademark of NXP B.V.

I2C-bus — logo is a trademark of NXP B.V.

Kinetis — is a trademark of NXP B.V.

Layerscape — is a trademark of NXP B.V.

Mantis — is a trademark of NXP B.V.

MIFARE — is a trademark of NXP B.V.

MOBILEGT — is a trademark of NXP B.V.

NTAG — is a trademark of NXP B.V.

Processor Expert — is a trademark of NXP B.V.

QorIQ — is a trademark of NXP B.V.

SafeAssure — is a trademark of NXP B.V.

SafeAssure — logo is a trademark of NXP B.V.

StarCore — is a trademark of NXP B.V.

Synopsys — Portions Copyright © 2021 Synopsys, Inc. Used with permission. All rights reserved.

Tower — is a trademark of NXP B.V.

UCODE — is a trademark of NXP B.V.

VortiQa — is a trademark of NXP B.V.

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 13 May 2022
Document identifier: AN13629