TB3172

How to Use XDMAC on Cortex™-M7 Microcontrollers to Implement Ping-Pong Buffering in Audio Applications

Introduction

Audio system applications can pose typical producer-consumer problems associated with a real-time system. The timing constraints include latency, sampling rate, sampling period and real-time response. To implement a solution to the audio time challenges, an application typically uses the Ping-Pong data buffer method. The Ping-Pong method uses two buffers: one buffer reads data from the source and another buffer is submitting the data to the destination. When the submission to the destination is complete, the roles of the two buffers are switched. The Ping-Pong is accomplished by modifying the index (pointers) to the buffers.

The following figure represents a typical USB Audio headphone application implementation where the PC (acting as a USB host) streams audio data to the Cortex™-M7 microcontroller (MCU) based audio subsystem. The application updates *Buffer1* with streamed in audio data from the USB. The XDMAC sets up the transfer of audio data to the CODEC (through the SSC) by configuring the address of *Buffer1* as the source address, and the address of the SSC transmit register (SSC_THR) as the destination address. When the data transmission is completed, the XDMAC switches the source address to *Buffer2* (which would be updated by the latest streamed in audio data). When the *Buffer2* data transmission is completed, the XDMAC switches the source address to *Buffer1*, which can be updated by the latest streamed audio data. This process of the XDMAC switching the source address from *Buffer1* to *Buffer2* continues as the data is streamed in and updated in the buffers while transmitting the data from the buffers to the CODEC to realize the audio playback at the headphone.

Figure 1. USB Audio Headphone Application

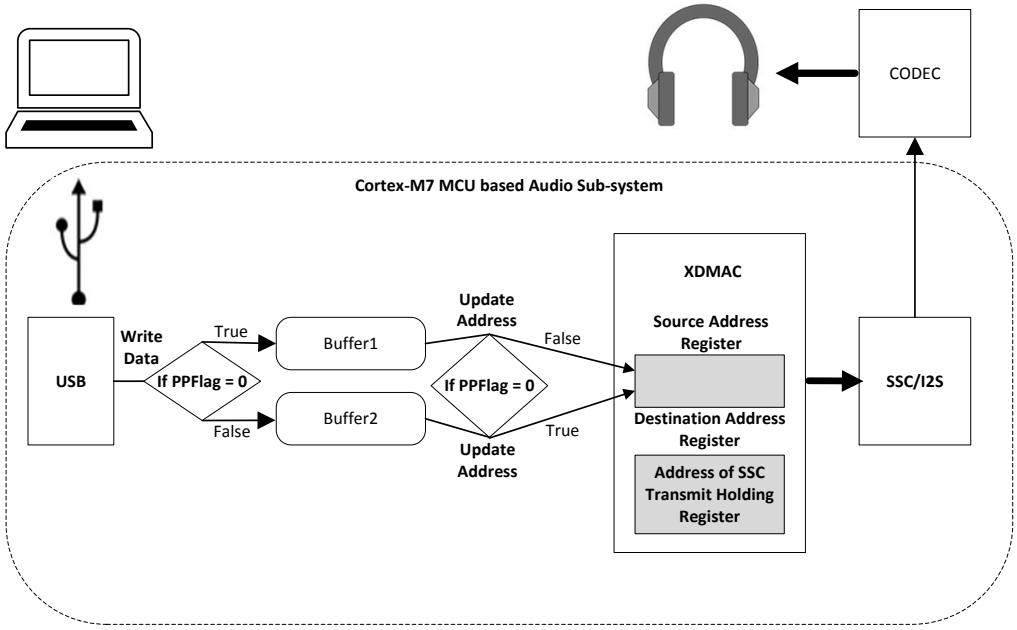


Table of Contents

1. Concept

The XDMAC peripheral provides support to implement the Ping-Pong buffering method through a multiple block data transfer feature. In multiple block data transfer, each block of data can be transferred with different configuration parameters such as, data source, destination address, data size, and trigger source. The XDMAC supports multiple block data transfer through the linked list. An implementation that uses a linked list feature requires the creation of linked list descriptors.

A linked list descriptor contains register settings to transfer a block of memory. When data is transferred using a linked list, the XDMAC fetches the first descriptor from the linked list, copies the register settings to channel registers, and performs the data transfer according to the configuration settings. When the first block transfer is completed, the XDMAC fetches the second descriptor from the linked list, copies the register settings to channel registers, and performs the data transfer according to the configuration settings. This process continues until the end of the linked list. Linked list descriptors are usually stored in internal data memory.

2. Solution

To implement Ping-Pong based audio data buffer handling in the USB Audio headphone application, build an XDMAC circular linked list consisting of two descriptors with the ‘View 1’ type. The ‘View 1’ type descriptor has four members, as shown in the following table.

Table 2-1. View 1 Type Linked List Descriptor

MBR_NDA		Next Descriptor Address
MBR_SA		Source Address
MBR_DA		Destination Address
MBR_UBC (Microblock Control)	**UBLEN**	Data length in microblocks
	NDE	Next Descriptor Enable
	NSEN	Next Descriptor Source Update
	NDEN	Next Descriptor Source Update
	NVIEW	Next Descriptor View Type

Note: In this application, although the ‘View 0’ type descriptor is sufficient, the ‘View 1’ type descriptor is chosen to demonstrate the use case where one item of control configuration changes (NSEN enabled) from one block to other while another item remains the same (NDEN disabled).

The following figure is a representation of the application buffers used in the Ping-Pong implementation

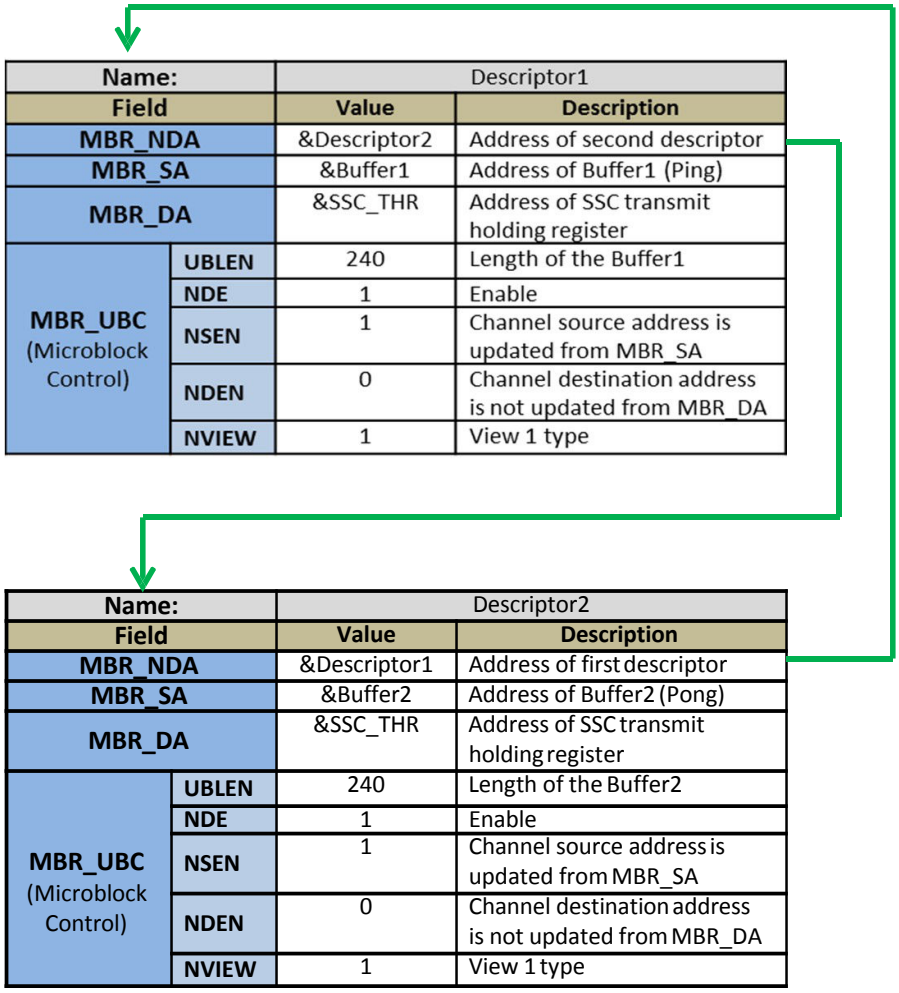
Figure 2-1. Ping-Pong Audio Data Buffers

Note: The application buffers *Buffer1* and *Buffer2* are shared between the XDMAC and the USB.

1. Typically, the application maintains the exclusive access to these buffers through a global flag, such as PPFlag, shown in Figure 1.
2. If XDMAC is transmitting *Buffer1*, it is the responsibility of the application to ensure that the *Buffer2* is updated with data from USB before the completion of *Buffer1*.

The following figure shows the XDMAC circular linked list with two descriptors to implement the Ping-Pong buffering.

Figure 2-2. Circular Linked List



Name:		Descriptor1	
Field		**Value**	**Description**
MBR_NDA		&Descriptor2	Address of second descriptor
MBR_SA		&Buffer1	Address of Buffer1 (Ping)
MBR_DA		&SSC_THR	Address of SSC transmit holding register
MBR_UBC (Microblock Control)	**UBLEN**	240	Length of the Buffer1
	NDE	1	Enable
	NSEN	1	Channel source address is updated from MBR_SA
	NDEN	0	Channel destination address is not updated from MBR_DA
	NVIEW	1	View 1 type

Name:		Descriptor2	
Field		**Value**	**Description**
MBR_NDA		&Descriptor1	Address of first descriptor
MBR_SA		&Buffer2	Address of Buffer2 (Pong)
MBR_DA		&SSC_THR	Address of SSC transmit holding register
MBR_UBC (Microblock Control)	**UBLEN**	240	Length of the Buffer2
	NDE	1	Enable
	NSEN	1	Channel source address is updated from MBR_SA
	NDEN	0	Channel destination address is not updated from MBR_DA
	NVIEW	1	View 1 type

1. The next descriptor fields (NDA, NDE, NSEN, NDEN and NVIEW) control the transfer behavior of the next descriptor, not the current descriptor.
   - *Descriptor1* controls the transfer behavior when the first descriptor is completed and the transfer of the second descriptor is to start
   - *Descriptor2* controls the transfer behavior when the second descriptor is completed and the transfer of the first descriptor is to start
2. The NSEN field in both of the linked list descriptors above is set as '1' to enable updates of the channel-specific source address register (XDMAC_CSAx) with the value in the MBR_SA field. This is because at the end of a block transfer, the XDMAC_CSAx would have an invalid address for the next transaction. MBR_SA contains the address of the buffer, which would be the next buffer made ready for transfer.
   - After the end of transmission of *Descriptor1*, XDMAC_CSAx is updated to contain the address of *Buffer2* (pong) from *Buffer1* (ping).
   - After the end of transmission of *Descriptor2*, XDMAC_CSAx is updated to contain the address of *Buffer1* (ping) from *Buffer2* (pong).

3. The NDEN field in both the linked list descriptors above is set as '0' to disable updates of the channel-specific destination address register (XDMAC_CDAx) with the value in the MBR_DA field. This is because at the end of a block transfer, the XDMAC_CDAx would still have a valid address for the next transaction.
   – After the end of transmission of *Descriptor1*, XDMAC_CDAx contains the address of SSC transmit holding register (SSC_THR) and subsequently is not updated.
   – After the end of transmission of *Descriptor2*, XDMAC_CDAx contains the address of SSC transmit holding register (SSC_THR) and subsequently is not updated.
4. To start the data transfer, enable the implemented linked list by directly writing to the channel specific next descriptor control (XDMAC_CNDCx) and address (XDMAC_CNDAx) registers with the values of the first descriptor (NDA, NDE, NSEN, NDEN and NVIEW).

The following figure shows example code to create the circular linked list.

Figure 2-3. Create Circular Linked List

```
typedef struct {
    /** Next Descriptor Address number. */
    uint32_t mbr_nda;
    /** Microblock Control Member. */
    uint32_t mbr_ubc;
    /** Source Address Member. */
    uint32_t mbr_sa;
    /** Destination Address Member. */
    uint32_t mbr_da;
} lld_view1;

/** Bits per slot */
#define BITS_BY_SLOT            (16)
/** Total number of buffers */
#define TOTAL_BUFFERS           2
/** Length of a buffer */
#define MAX_DMA_SIZE            240
/** XDMA Descriptor */
static lld_view1 linklist_write[TOTAL_BUFFERS];
/** Audio Buffer */
static uint16_t Buffer[TOTAL_BUFFERS][MAX_DMA_SIZE*(BITS_BY_SLOT / 8)];

  for(i = 0; i < TOTAL_BUFFERS; i++)
  {
      linklist_write[i].mbr_ubc = XDMAC_UBC_NVIEW_NDV1
                                | XDMAC_UBC_NDE_FETCH_EN
                                | XDMAC_UBC_NSEN_UPDATED
                                | XDMAC_UBC_NDEN_UNCHANGED
                                | XDMAC_CUBC_UBLEN(MAX_DMA_SIZE);
      linklist_write[i].mbr_sa = (uint32_t)(&Buffer[i]);
      linklist_write[i].mbr_da = (uint32_t)&(SSC->SSC_THR);
      if ( i == (TOTAL_BUFFERS - 1 ))
      {
          linklist_write[i].mbr_nda = (uint32_t)&linklist_write[0];
      }
      else
      {
          linklist_write[i].mbr_nda = (uint32_t)&linklist_write[i+1];
      }
  }
```

After initializing the linked list as previously described, and once the corresponding XDMAC channel is enabled, the XDMAC places the contents of *Buffer1* onto the SSC transmit register (SSC_THR) according to the configuration parameters. When the *Buffer1* transmission is completed, the XDMAC switches to *Descriptor2* in the linked list. The source address of the audio data in *Descriptor2* points to *Buffer2*. Once transmission completes, as with earlier descriptor, the XDMAC moves back to the *Descriptor1* in the list. The XDMAC switches between the two descriptors (implicitly between the two Ping-Pong audio buffers *Buffer1* and *Buffer2*) while transmitting the audio data through the SSC to the CODEC.

3. Relevant Resources

- Usage of XDMAC on SAM S/SAM E/SAM V
  http://www.atmel.com/Images/Atmel-42761-Usage-of-XDMAC-on-SAMS-SAME-SAMV_ApplicationNote_AT17417.pdf
- Synchronous Serial Controller (SSC) of SAMV71 Devices to Output an Audio Stream Through the On-board WM8904 CODEC
  http://asf.atmel.com/docs/latest/sam.components.audio.codec.wm8904.example.samv71_xplained_ultra/html/index.html
- Synchronous Serial Controller (SSC) of SAM3S Devices to Output an Audio Stream Through the On-board WM8731 CODEC
  http://asf.atmel.com/docs/latest/sam.components.audio.codec.wm8731.example.sam3x_ek/html/index.html

The Microchip Web Site

Microchip provides online support via our web site at http://www.microchip.com/. This web site is used as a means to make files and information easily available to customers. Accessible by using your favorite Internet browser, the web site contains the following information:

- **Product Support** – Data sheets and errata, application notes and sample programs, design resources, user's guides and hardware support documents, latest software releases and archived software
- **General Technical Support** – Frequently Asked Questions (FAQ), technical support requests, online discussion groups, Microchip consultant program member listing
- **Business of Microchip** – Product selector and ordering guides, latest Microchip press releases, listing of seminars and events, listings of Microchip sales offices, distributors and factory representatives

Customer Change Notification Service

Microchip's customer notification service helps keep customers current on Microchip products. Subscribers will receive e-mail notification whenever there are changes, updates, revisions or errata related to a specified product family or development tool of interest.

To register, access the Microchip web site at http://www.microchip.com/. Under "Support", click on "Customer Change Notification" and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Field Application Engineer (FAE)
- Technical Support

Customers should contact their distributor, representative or Field Application Engineer (FAE) for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in the back of this document.

Technical support is available through the web site at: http://www.microchip.com/support

Microchip Devices Code Protection Feature

Note the following details of the code protection feature on Microchip devices:

- Microchip products meet the specification contained in their particular Microchip Data Sheet.
- Microchip believes that its family of products is one of the most secure families of its kind on the market today, when used in the intended manner and under normal conditions.
- There are dishonest and possibly illegal methods used to breach the code protection feature. All of these methods, to our knowledge, require using the Microchip products in a manner outside the operating specifications contained in Microchip's Data Sheets. Most likely, the person doing so is engaged in theft of intellectual property.
- Microchip is willing to work with the customer who is concerned about the integrity of their code.

- Neither Microchip nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as “unbreakable.”

Code protection is constantly evolving. We at Microchip are committed to continuously improving the code protection features of our products. Attempts to break Microchip’s code protection feature may be a violation of the Digital Millennium Copyright Act. If such acts allow unauthorized access to your software or other copyrighted work, you may have a right to sue for relief under that Act.

Legal Notice

Trademarks

ISBN: 978-1-5224-2385-0

Quality Management System Certified by DNV

ISO/TS 16949

Microchip received ISO/TS-16949:2009 certification for its worldwide headquarters, design and wafer fabrication facilities in Chandler and Tempe, Arizona; Gresham, Oregon and design centers in California and India. The Company's quality system processes and procedures are for its PIC® MCUs and dsPIC® DSCs, KEELOQ® code hopping devices, Serial EEPROMs, microperipherals, nonvolatile memory and analog products. In addition, Microchip's quality system for the design and manufacture of development systems is ISO 9001:2000 certified.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Fax: 480-792-7277 Technical Support: http://www.microchip.com/support Web Address: www.microchip.com **Atlanta** Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455 **Austin, TX** Tel: 512-257-3370 **Boston** Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088 **Chicago** Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075 **Dallas** Addison, TX Tel: 972-818-7423 Fax: 972-818-2924 **Detroit** Novi, MI Tel: 248-848-4000 **Houston, TX** Tel: 281-894-5983 **Indianapolis** Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380 **Los Angeles** Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800 **Raleigh, NC** Tel: 919-844-7510 **New York, NY** Tel: 631-435-6000 **San Jose, CA** Tel: 408-735-9110 Tel: 408-436-4270 **Canada - Toronto** Tel: 905-695-1980 Fax: 905-695-2078	**Asia Pacific Office** Suites 3707-14, 37th Floor Tower 6, The Gateway Harbour City, Kowloon **Hong Kong** Tel: 852-2943-5100 Fax: 852-2401-3431 **Australia - Sydney** Tel: 61-2-9868-6733 Fax: 61-2-9868-6755 **China - Beijing** Tel: 86-10-8569-7000 Fax: 86-10-8528-2104 **China - Chengdu** Tel: 86-28-8665-5511 Fax: 86-28-8665-7889 **China - Chongqing** Tel: 86-23-8980-9588 Fax: 86-23-8980-9500 **China - Dongguan** Tel: 86-769-8702-9880 **China - Guangzhou** Tel: 86-20-8755-8029 **China - Hangzhou** Tel: 86-571-8792-8115 Fax: 86-571-8792-8116 **China - Hong Kong SAR** Tel: 852-2943-5100 Fax: 852-2401-3431 **China - Nanjing** Tel: 86-25-8473-2460 Fax: 86-25-8473-2470 **China - Qingdao** Tel: 86-532-8502-7355 Fax: 86-532-8502-7205 **China - Shanghai** Tel: 86-21-3326-8000 Fax: 86-21-3326-8021 **China - Shenyang** Tel: 86-24-2334-2829 Fax: 86-24-2334-2393 **China - Shenzhen** Tel: 86-755-8864-2200 Fax: 86-755-8203-1760 **China - Wuhan** Tel: 86-27-5980-5300 Fax: 86-27-5980-5118 **China - Xian** Tel: 86-29-8833-7252 Fax: 86-29-8833-7256	**China - Xiamen** Tel: 86-592-2388138 Fax: 86-592-2388130 **China - Zhuhai** Tel: 86-756-3210040 Fax: 86-756-3210049 **India - Bangalore** Tel: 91-80-3090-4444 Fax: 91-80-3090-4123 **India - New Delhi** Tel: 91-11-4160-8631 Fax: 91-11-4160-8632 **India - Pune** Tel: 91-20-3019-1500 **Japan - Osaka** Tel: 81-6-6152-7160 Fax: 81-6-6152-9310 **Japan - Tokyo** Tel: 81-3-6880- 3770 Fax: 81-3-6880-3771 **Korea - Daegu** Tel: 82-53-744-4301 Fax: 82-53-744-4302 **Korea - Seoul** Tel: 82-2-554-7200 Fax: 82-2-558-5932 or 82-2-558-5934 **Malaysia - Kuala Lumpur** Tel: 60-3-6201-9857 Fax: 60-3-6201-9859 **Malaysia - Penang** Tel: 60-4-227-8870 Fax: 60-4-227-4068 **Philippines - Manila** Tel: 63-2-634-9065 Fax: 63-2-634-9069 **Singapore** Tel: 65-6334-8870 Fax: 65-6334-8850 **Taiwan - Hsin Chu** Tel: 886-3-5778-366 Fax: 886-3-5770-955 **Taiwan - Kaohsiung** Tel: 886-7-213-7830 **Taiwan - Taipei** Tel: 886-2-2508-8600 Fax: 886-2-2508-0102 **Thailand - Bangkok** Tel: 66-2-694-1351 Fax: 66-2-694-1350	**Austria - Wels** Tel: 43-7242-2244-39 Fax: 43-7242-2244-393 **Denmark - Copenhagen** Tel: 45-4450-2828 Fax: 45-4485-2829 **Finland - Espoo** Tel: 358-9-4520-820 **France - Paris** Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79 **France - Saint Cloud** Tel: 33-1-30-60-70-00 **Germany - Garching** Tel: 49-8931-9700 **Germany - Haan** Tel: 49-2129-3766400 **Germany - Heilbronn** Tel: 49-7131-67-3636 **Germany - Karlsruhe** Tel: 49-721-625370 **Germany - Munich** Tel: 49-89-627-144-0 Fax: 49-89-627-144-44 **Germany - Rosenheim** Tel: 49-8031-354-560 **Israel - Ra'anana** Tel: 972-9-744-7705 **Italy - Milan** Tel: 39-0331-742611 Fax: 39-0331-466781 **Italy - Padova** Tel: 39-049-7625286 **Netherlands - Drunen** Tel: 31-416-690399 Fax: 31-416-690340 **Norway - Trondheim** Tel: 47-7289-7561 **Poland - Warsaw** Tel: 48-22-3325737 **Romania - Bucharest** Tel: 40-21-407-87-50 **Spain - Madrid** Tel: 34-91-708-08-90 Fax: 34-91-708-08-91 **Sweden - Gothenberg** Tel: 46-31-704-60-40 **Sweden - Stockholm** Tel: 46-8-5090-4654 **UK - Wokingham** Tel: 44-118-921-5800 Fax: 44-118-921-5820