



APB-音频播放器

2024.08.31

一 修订

版本	日期	编辑人	内容
1.00	2024.08.31	陈家耀	创建了第一个正式版本
1.01	2024.09.18	陈家耀	指出“可以更改音频源”

二 简介和特性

APB-音频播放器是一个带有 APB 从接口的音频播放器，可直接挂载在 APB 总线上充当音频播放器外设。本 IP 简单易用、资源消耗少，具有以下特性：

- 使用 SPI-Flash 作为音频数据存储
- 运行时可配置的音频采样率，支持 8KHz、11.025KHz、22.05KHz、24KHz、32KHz、44.1KHz、47.25KHz、48KHz
- 12 位 DAC 输出，支持 8 位或 16 位音频数据量化精度
- 单声道音频播放器
- 支持音频播放中止

APB-音频播放器的组成如图 2-1 所示，它由 APB 寄存器接口、Flash DMA、SPI 控制器和音频 DAC 控制器四部分组成。Flash DMA 接收来自 APB 寄存器接口的 DMA 请求，生成 SPI-Flash 读命令。SPI 控制器从发送 fifo 取出待传输字节，驱动 SPI 接口以完成 SPI 传输。音频 DAC 控制器根据用户指定的音频采样率，周期性地从音频 fifo 取数据，给出 12 位 DAC 输出（若音频 fifo 空则输出 0）。

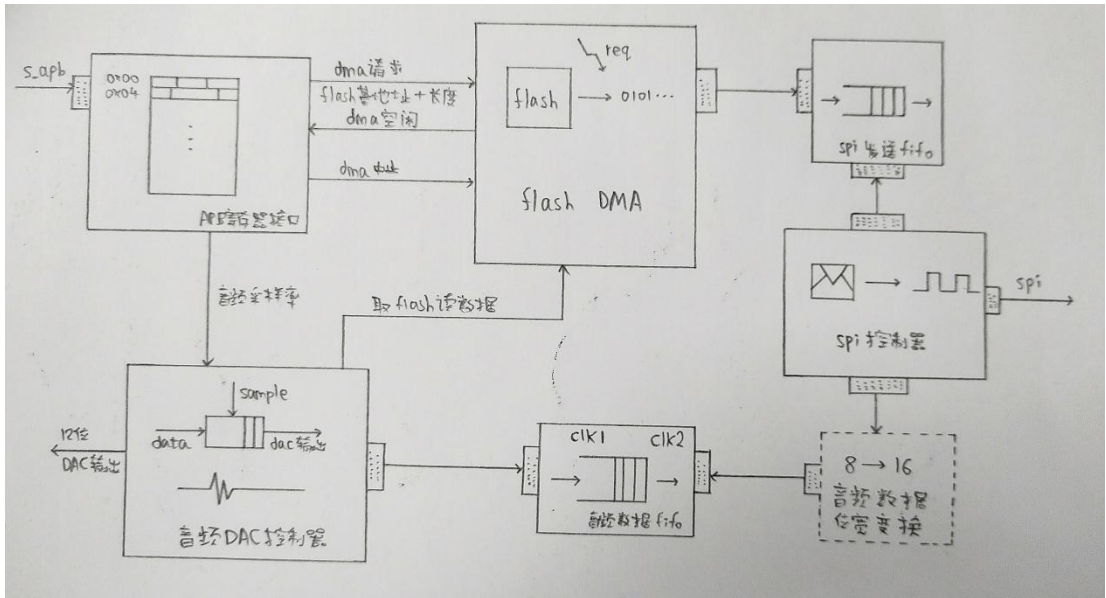


图 2-1 APB-音频播放器组成框图

APB-音频播放器使用独立的 APB 配置时钟（amba_clk、amba_resetrn）和音频时钟（audio_clk、audio_resetrn）。APB 从接口和 SPI 接口使用 APB 配置时钟，而 DAC 输出使用音频时钟，请注意区分。

三 IP 功能

APB-音频播放器是通用的音频播放器外设，能从 SPI-Flash 取出音频数据，根据设定的音频采样率给出 12 位 DAC 输出。其功能描述如下：

- 1、**开始播放一段音频**。向 Flash DMA 提交请求后，本 IP 将自动从 SPI-Flash 读取音频数据，数据被存入音频 fifo 后将以符合音频采样率的速度取出并产生 DAC 输出。
- 2、**中止一段音频**。若当前 Flash DMA 有处理中的 Flash 读请求，用户可发出中止请求以停止播放当前音频。若当前没有 Flash 读请求，则中止请求被忽略。
- 3、**配置音频采样率**。用户可通过 APB 寄存器接口配置音频采样率，支持 8KHz、11.025KHz、22.05KHz、24KHz、32KHz、44.1KHz、47.25KHz、48KHz。

四 IO 描述

表 4-1 APB-音频播放器 IO 表

端口名	方向	位宽	含义
APB 从接口时钟和复位			
amba_clk	input	1	时钟
amba_resetn	input	1	复位，低有效
音频时钟和复位			
audio_clk	input	1	时钟
audio_resetn	input	1	复位，低有效
APB 从接口			
paddr	input	32	APB 从机地址
psel	input	1	APB 从机片选
penable	input	1	APB 从机传输使能
pwrite	input	1	APB 从机读写类型
pwrdata	input	32	APB 从机写数据
pready_out	output	1	APB 从机传输完成，固定为 1
prdata_out	output	32	APB 从机读数据
pslverr_out	output	1	APB 从机传输错误，固定为 0
SPI 主接口			
audio_flash_spi_ss	output	1	SPI 片选
audio_flash_spi_sck	output	1	SPI 时钟
audio_flash_spi_mosi	output	1	SPI 主机输出
audio_flash_spi_miso	input	1	SPI 主机输入
12 位 DAC 输出			
audio_dac_out	output	12	DAC 输出

五 可配置参数描述

表 5-1 APB-GPIO 可配置参数表

配置参数名	含义	可取值
audio_clk_freq	音频时钟频率，以 Hz 计	浮点数， $\geq 1 \times 10^6$
init_audio_sample_rate	复位时的音频采样率	[0, 7]内的整数
audio_qtz_width	音频数据量化精度	8 16
audio_sample_rate_fixed	音频采样率是否固定	"true" "false"
simulation_delay	仿真延时, 可用于仿真时模拟 D 到 Q 延迟	0.1f~100.0f

表 5-2 音频采样率对照表

编号	采样率
0	8KHz
1	11.025KHz
2	22.05KHz
3	24KHz
4	32KHz
5	44.1KHz
6	47.25KHz
7	48KHz

六 应用指南

6.1 RTL 设计指南

APB-音频播放器是标准的 APB 外设，请将 APB-音频播放器挂载在 APB 总线上使用，典型情况是挂载在 AXI-APB 桥或 AHB-APB 桥上作为一个 APB 从机，如图 6-1-1 所示。关于 AXI-APB 桥或 AHB-APB 桥，请参见 UG200。

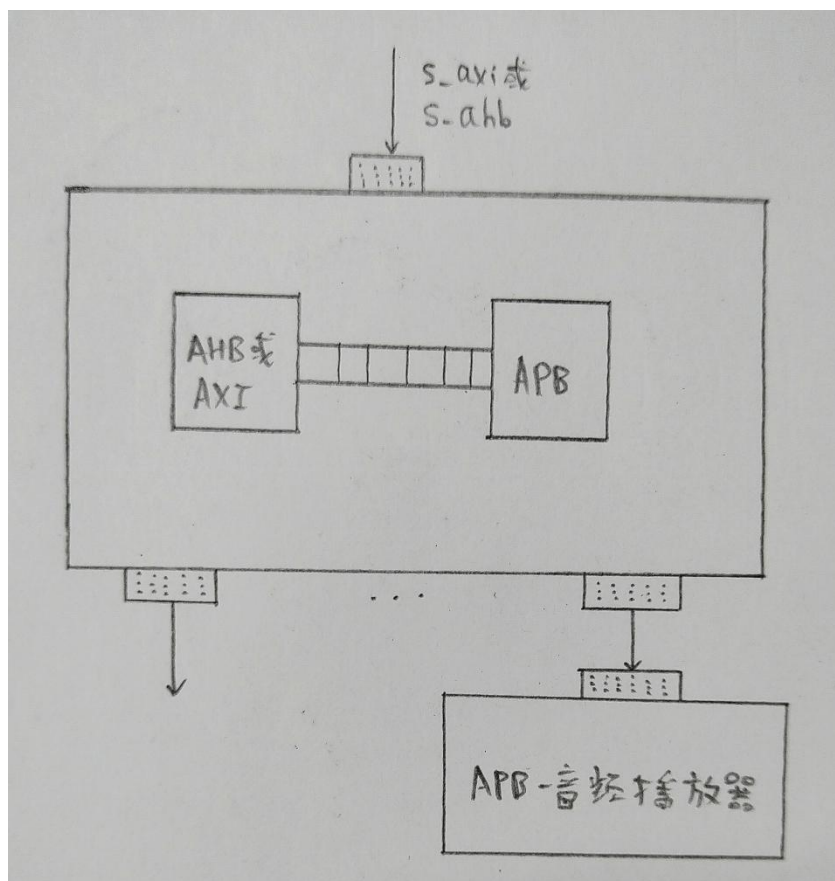


图 6-1-1 APB-音频播放器应用图

APB-音频播放器允许用户修改音频源，只需要重新连接音频 fifo 的写端口，删除 flash DMA、spi 发送 fifo 和 spi 控制器。

APB-音频播放器以 flash 作为音频数据存储，flash 型号为 W25QXX，其读命令的格式为：

8'h03

基地址[23:16]

基地址[15:8]

基地址[7:0]

读数据#0

...

读数据#N-1

如果您使用的 flash 型号不是 W25QXX，可以修改 flash_dma.v 中的 SPI 传输

命令产生逻辑。

如果您不希望使用 flash 作为音频数据存储器，那么不应该使用 flash_dma.v 和 std_spi_tx_rx.v，并重新设计一个音频存储器 DMA 和音频数据传输接口控制器。

本 IP 所提供的同步 FIFO 的顶层 RTL 文件为 ram_fifo_wrapper.v，由于 fifo 使用到的 RAM 可能与器件类型有关，必要时请根据设计要求进行替换。本 IP 所提供的异步 FIFO 的顶层 RTL 文件为 async_fifo_with_ram.v，由于 fifo 使用到的 RAM 可能与器件类型有关，必要时请根据设计要求进行替换。

6.2 软件编程指南

6.2.1 软件驱动 API

1、类型定义

- ApbAudio 结构体（APB-音频播放器外设结构体）

ApbAudioHd* hardware: APB-音频播放器寄存器接口结构体指针，映射到音频采样率外设的寄存器接口

- ApbAudioHd 结构体（APB-音频播放器寄存器接口结构体）

uint32_t flash_rd_baseaddr: APB-音频播放器寄存器接口偏移地址 0x00，音频在 flash 中的基地址

uint32_t flash_rd_bytes_n: APB-音频播放器寄存器接口偏移地址 0x04，音频字节数

uint32_t sample_rate: APB-音频播放器寄存器接口偏移地址 0x08，音频采样率类型

uint32_t flash_dma_cs: APB-音频播放器寄存器接口偏移地址 0x0C，flash-dma 控制

uint32_t err_flag_vec: APB-音频播放器寄存器接口偏移地址 0x10，错误标志向量

2、宏函数

- **__APB_AUDIO_PLAYER_ABORT_DMA(audio_player)**
简介：中止播放音频
参数：**audio_player** APB-音频播放器（结构体指针）
返回值：无
- **__IS_APB_AUDIO_PLAYER_IDLE(audio_player)**
简介：判断 flash-dma 是否空闲
参数：**audio_player** APB-音频播放器（结构体指针）
返回值：flash-dma 是否空闲
- **__APB_AUDIO_PLAYER_GET_ERR_FLAG(audio_player)**
简介：获取错误标志向量
参数：**audio_player** APB-音频播放器（结构体指针）
返回值：错误标志向量

- **__APB_AUDIO_PLAYER_CLC_ERR_FLAG(audio_player)**
简介：清除错误标志向量
参数：**audio_player** APB-音频播放器（结构体指针）
返回值：无

3、宏定义

- 音频采样率类型
 - AUDIO_SAMPLE_RATE_8000** 音频采样率 8KHz
 - AUDIO_SAMPLE_RATE_11025** 音频采样率 11.025KHz
 - AUDIO_SAMPLE_RATE_22050** 音频采样率 22.05KHz
 - AUDIO_SAMPLE_RATE_24000** 音频采样率 24KHz
 - AUDIO_SAMPLE_RATE_32000** 音频采样率 32KHz
 - AUDIO_SAMPLE_RATE_44100** 音频采样率 44.1KHz
 - AUDIO_SAMPLE_RATE_47250** 音频采样率 47.25KHz
 - AUDIO_SAMPLE_RATE_48000** 音频采样率 48KHz
- 错误类型掩码
 - ERR_SPI_TX_FIFO_WT_OVF_MASK SPI** 控制器发送 fifo 写溢出
 - ERR_SPI_RX_FIFO_WT_OVF_MASK SPI** 控制器接收 fifo 写溢出

4、函数

- **void init_apb_audio_player(ApbAudio* audio_player, uint32_t baseaddr);**
简介：初始化 APB-音频播放器
参数：**audio_player** APB-音频播放器（结构体指针）
 base_addr APB-音频播放器外设基地址
返回值：无
- **void apb_audio_player_set_audio_sample_rate(ApbAudio* audio_player, uint8_t audio_sample_rate);**
简介：设置音频采样率
参数：**audio_player** APB-音频播放器（结构体指针）
 audio_sample_rate 音频采样率类型
返回值：无
- **int apb_audio_player_start_dma(ApbAudio* audio_player, uint32_t flash_rd_baseaddr, uint32_t flash_rd_bytes_n)**
简介：启动 flash-dma，开始播放一段音频
参数：**audio_player** APB-音频播放器（结构体指针）
 flash_rd_baseaddr 音频在 flash 中的基地址
 flash_rd_bytes_n 音频字节数
返回值：是否成功

6.2.2 软件编程示例

在本示例中，音频时钟频率为 20MHz，音频数据量化精度为 8 位，音频采样率为 22.05KHz，从 flash 基地址 0x0000_0000 开始写入了 72 秒的音频数据。音频数据见 `firmware/examples/music.bin`。flash 编程算法可参见 `apb_spi` 的文档（UG106）。

开始运行程序后，自动播放音乐《安妮的仙境》，持续 10s 后停止。

```
1.  /*****
   *****/
2.  APB-音频播放器示例代码
3.  @brief 播放音乐 10s 然后停止
4.  @attention 请根据硬件平台更换与延迟(delay)相关的 API
5.  @date 2024/08/31
6.  @author 陈家耀
7.  @eidt 2024/08/31 1.00 创建了第一个正式版本
8.  *****/
   *****/
9.
10. #include "../apb_audio_player.h"
11. #include "../delay.h"
12.
13. //////////////////////////////////////
14.
15. #define APB_AUDIO_PLAYER_BASEADDR 0x40000000 // APB-音频播放器外设基地址
16.
17. #define MUSIC_FLASH_BASEADDR 0 // 音乐在 flash 中的基地址
18. #define MUSIC_LEN (72 * 22050) // 音乐长度(以字节计)
19.
20. //////////////////////////////////////
21.
22. static ApbAudio audio_player; // 音频播放器外设句柄
23.
24. //////////////////////////////////////
25.
26. void apb_audio_player_simple_example(void){
27.     // 初始化音频播放器
28.     init_apb_audio_player(&audio_player, APB_AUDIO_PLAYER_BASEADDR);
29.
30.     // 设置音频采样率为 22.05KHz
31.     apb_audio_player_set_audio_sample_rate(&audio_player, AUDIO_SAMPLE_RATE_22050);
32.
33.     // 开始播放音乐
34.     apb_audio_player_start_dma(&audio_player, MUSIC_FLASH_BASEADDR, MUSIC_LEN);
35.
```

```
36.     delay_ms(10 * 1000); // 延迟 10s
37.
38.     __APB_AUDIO_PLAYER_ABORT_DMA(&audio_player); // 停止播放音乐
39.
40.     while(1);
41. }
```