文档编号： PROCHIP_GD00_DRV_05

GD00 驱动开发手册

AC97 驱动和 MP3 文件的播放

南京博芯电子技术有限公司

版本说明

版本号	日期	编写人	描述
0001	2005-5-25	张兵	

本文档适用于 GD00 开发板

参考文档

文档名	文档编号
《东芯IV SEP3203F50用户手册》	PROCHIP_SEP3203_USER_01
《ASIX OS 中基于东芯 IV SEP3203F50 的 MP3 解码库使用说明》	PROCHIP_SEP3203_DEV_07

目录

1 说明

本项目为东芯 IV SEP3203F50 处理器中 MP3 应用的演示程序，可以作为基本硬件功能测试，也可以作为开发 MP3 解码播放系统的模板。项目文件夹中的物理布局参照文档 readme.txt。本文档主要介绍 MP3 播放的接口程序，主要针对最常见的 44.1KHZ 采样率的 MP3 歌曲解码播放。

2 模块说明

MP3播放主要使用SEP3203F50的AC97、DMA、MMA、eSRAM等片内模块以及片外音频Codec—UCB1400。硬件系统的简化框图如下图所示：

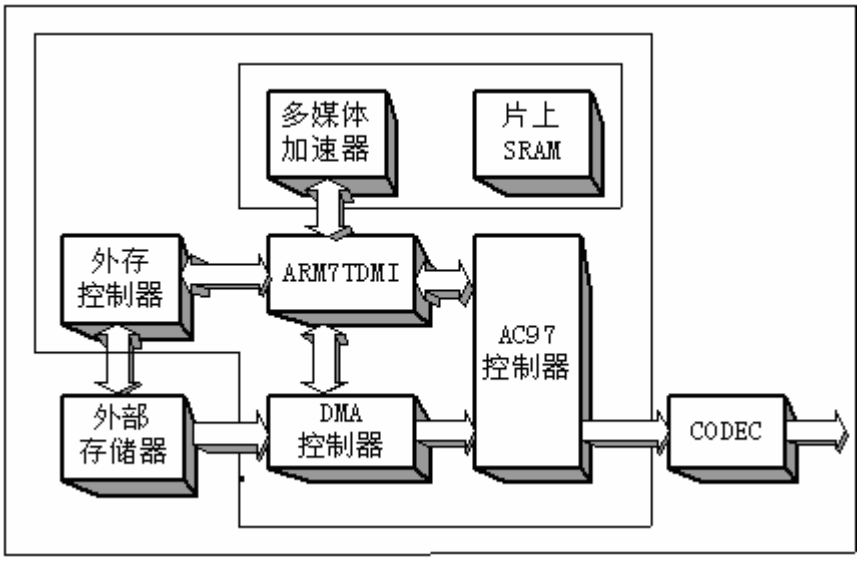

图1 MP3子系统硬件框图

根据图 1，在 MP3 的应用中，使用到的 SEP3203F50 片内各模块基本功能如下所述：
外部存储器接口（EMI）：支持 CPU 对 SRAM，SDRAM 和 FLASH 等内存的访问；
片上高速 SRAM（eSRAM）：利用自身零等待的特点优化耗时的核心代码以提高系统性能；
多媒体加速模块（MMA）：通过硬件实现向量乘法以支持 MP3 实时解码，提高系统性能；
音频控制器 AC97：为音频处理提供与 AC97 标准兼容的输入输出支持；
DMA 控制器：实现无需内核干涉的大数据量快速传送。

2.1 AC97

AC'97 控制器（Audio Codec'97 Controller ）主要是通过AC-Link 实现与AC'97 Codec 的通讯。在SEP3203F50里，主要只实现AC'97 Controller 左右声道到Codec 的数据传送，以及Codec到Microphone 或者左右通道（三选一）到AC'97 Controller 数据传输。AC'97 控制器与Codec 之间的通讯遵循Intel AC'97 (Audio Codec'97)协议。具有以下功能：

- 支持AC'97 Controller 左右声道到Codec 的数据传送
- 录音通道可配，支持Codec Microphone 和左右单声道（三选一）到AC'97 Controller 数据传输
- 具有DMA功能，可以向DMA 控制器请求DMA 操作，通过DMA 实现数据的传输
- 支持固定采样率和可调采样率，最大采样率为48KHz

2.2 DMA

直接存储器访问（DMA）方式，是一种完全由硬件执行数据交换的工作方式。在这种方式中，DMA 控制器从CPU 完全接管对总线的控制，数据交换不经过CPU 而直接在存储器之间以及存储器和外设之间进行。DMA 方式一般用于高速传送成组的数据。DMA 控制器(DMAC)产生地址、数据信号和控制信号。在传输过程中DMAC 可以自动增加地址，对传送的字的个数计数，并且以中断方式向CPU 报告传送操作的结束，提供如下功能：

- 6个DMA 通道，均可支持双向传输。
- 提供Single DMA 和Burst DMA 请求模式。Burst 传输的尺寸可编程配置。
- 提供存储器到外设，外设到存储器，存储器到存储器的数据传输。
- 硬件DMA 通道优先级。
- AHB slave 编程接口。
- 支持目的地址或源地址的递增(incremental)或非递增(non-incremental)产生方式。
- 可编程的DMA burst 尺寸。
- 16x4 Bytes 内部数据FIFO。
- 支持8,16 和32-bit 数据宽度的传输。
- 两个可屏蔽中断请求：DMA 错误中断和DMA 传输完成中断请求。

2.3 MMA

在MP3 等音视频中需要许多数字信号处理运算法，这些运算法则包括相关性，量化，子带合成和DCT 操作。在系统的执行中，这些操作占用了大多数的操作时间。多媒体加速器采用硬件固化一些使用频繁耗时的基本算法，提高音视频系统运行速度。

2.4 eSRAM

片上SRAM（eSRAM）的存取速度非常快，所以对一般的SoC 存储系统性能有很大的优化作用。片上SRAM 可以提供比外存高许多的性能（0.89MIPS/MHz），用户可以充分地利用这20K 的eSRAM，达到优化系统性能的目的，比如将音视频核心代码、中断处理程序、操作系统内核等放入其中。具有以下特点：

- 20Kbytes 快速eSRAM
- 单周期数据读写
- 支持8/16/32 位操作
- 可映射至零地址，支持从NAND FLASH 初始化
- 运算性能较高（0.89MIPS/MHz）

3 驱动介绍

3.1 接口系统概述

MP3 解码播放的核心接口程序为函数 play_mp3（），其基本流程如图 2 所示。

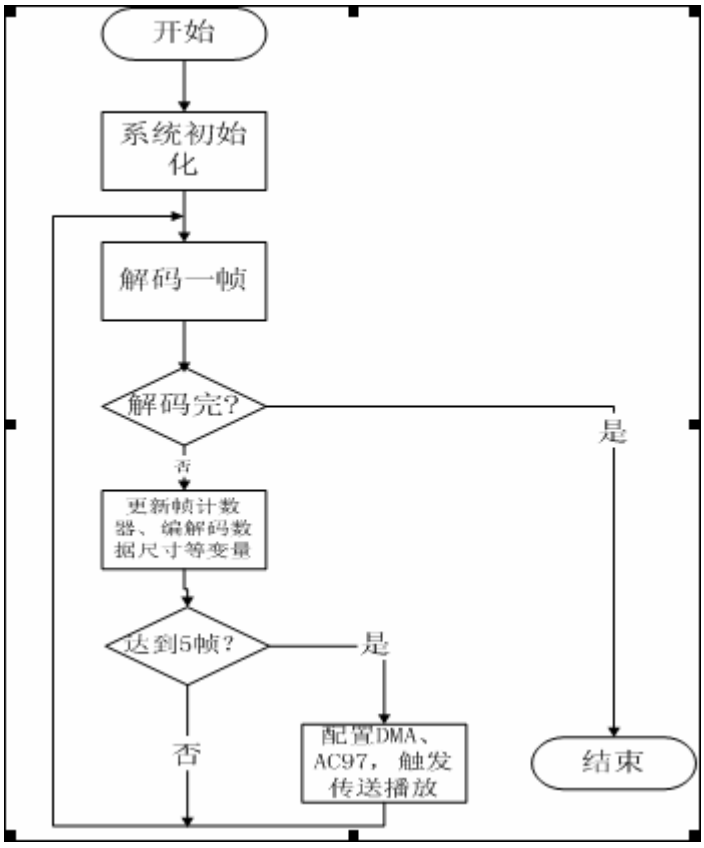

图 2 MP3 Demo 的基本流程

对照文件 main.c 中函数 play_mp3 ()来看此图的各个步骤：

① 系统初始化包括：配置时钟 config_pmu（）、配置中断控制器 init_intc（）、DMA 传送边界对齐、MP3 解码器结构体初始化 InitMP3（）、以及相关变量的初始化；

② 进入 while()循环后，解码一帧 MP3 数据 decodeMP3(&gmp, &encbuf[penc], lenenc, &decbuf[pdec], lendec, &incdec)；

③ 解码完，判断是否解完了所有 Mp3 数据，如果解完跳出循环，否则更新有关参数并继续解码；如果解码达到 5 帧，则配置 DMA、AC97 开始播放音乐。

注意：一旦配置过 DMA 和 AC97 后，就会以中断方式传送音乐数据(PCM 码)。在解码过程中，PCM 码由解码缓冲区向 AC97 输出 FIFO 的搬运由 DMA 负责，因此是解码和播放并发进行；只是每次 DMA 完成一次传送，会进入中断处理函数 ac97_handler（），再次配置 DMA，触发新的传送以连续播放。

3.2 函数说明

① MP3 播放的 demo 函数 int main(void)

该函数比较简单，演示了如何调用接口函数 play_mp3（）。
函数执行流程为：配置系统时钟 — 播放 MP3 — 返回

② MP3 播放的接口函数 int play_mp3(char * mp3_ptr)

mp3_ptr：存放待播放的 MP3 文件的内存区首地址，目前 demo 中以 char 型数组形式存放了一首测试歌曲，数据截取自 MP3 文件“马不停蹄的忧伤”。用户也可以根据自身需要在指定内存区导入选定的 MP3 文件，并把该首地址作为入口参数调用函数 play_mp3（）即可播放音乐了。

③MP3 解码的接口函数 int decodeMP3(struct mpstr *mp, char *in, int isize, char *out, int osize,

int *pcmsize)；
decodeMP3 是上层应用程序和 mp3 解码库之间的的接口函数，上层应用程序要调用解码库只要调用此函数即可，每调用一次，就解码一帧数据。如果用户要根据自身需要设计更加完善的 MP3 解码播放系统，就需要明确知道该函数用法
struct mpstr *mp：mp3 解码库系统数据结构，与 InitMP3 函数的入口函数用的是同一个全局变量；
char *in：mp3 解码输入 buffer 地址；
int isize：被解码数据的剩余总长度，若此参数值小于 4 则接口函数直接返回；
char *out ：mp3 解码输出 buffer 地址；
int osize：解出的数据的总长度，如果此参数值小于 0 则接口函数直接返回；
int *pcmsize：存放被解出的 pcm 码的长度；
函数返回值：返回值为当前被解码的帧长度。

4 驱动使用

该驱动的使用步骤如下；

- 运行 sys_init_emi\sys_init_emi\sys_init_emi.mcp 工程，进行 GD00 系统初始化。
- 根据需要配置系统时钟；
- 确定 MP3 文件存放在内存区的首地址 mp3_ptr，并调用函数 play_mp3(mp3_ptr)；
- 根据以上步骤编写代码，编译代码直至没有错误；
- 打开 GD00 硬件系统，并启动 Mult-ICE Server 软件，正确识别到硬件系统为 ARM7TDMI；
- 运行该工程，就可以播放存放在以 mp3_ptr 为首地址的 MP3 音乐文件了；

注意：本 Demo 已经用数组 encbuf[]存放了一首 MP3 歌曲的数据用于演示，截取自“马不停蹄的忧伤”，因此在函数 main()中调用为 play_mp3(&encbuf[0])，直接运行该工程就可以播放该歌曲。

5 附录 代码目录项说明

① 工程文件路径：\mp3 demo\build\mp3demo.mcp

② 物理文件夹说明：

boot：存放系统启动初始化文件 init_ice.s 和中断向量表文件 boot_gfd.s
misc：存放堆栈配置、内存映像等文件
include：存放有关寄存器、变量、函数声明以及宏定义
hardware；存放有关硬件配置文件
mp3lib：存放 MP3 解码算法的库函数文件

③ 重要的文件说明

main.c：MP3 播放的 demo 文件
HA_AC97.c：用于 MP3 播放的硬件配置代码等
mbtt.data：被解码播放的 MP3 文件数据，截取自“马不停蹄的忧伤”，存放形式为数组格式

④ 工程 mp3demo.mcp 打开后，各文件被组织到相应于物理文件夹的 Group 中