

HPM I2S DMA CHAIN 开发案例

目录

| | |
|-----------------------|----|
| 1. 简介 | 4 |
| 2. 整体方案 | 5 |
| 3. I2S DMA 接收功能 | 6 |
| 3.1 DMA 链式传输 | 6 |
| 3.2 CPU 处理数据 | 7 |
| 4. I2S DMA 发送功能 | 7 |
| 5. 测试验证 | 8 |
| 5.1 实验准备 | 8 |
| 5.2 测试验证 | 9 |
| 6. 总结 | 12 |

版本：

| 日期 | 版本号 | 说明 |
|------------|-----|----|
| 2024-09-20 | 1.1 | 初版 |
| | | |

1.简介

HPM6000 系列 MCU 是来自上海先楫半导体科技有限公司的高性能实时 RISC-V 微控制器，为工业自动化及边缘计算应用提供了极大的算力、高效的控制能力。上海先楫半导体目前已经发布了如 HPM6700/6400、HPM6300、HPM6200 等多个系列的高性能微控制器产品。

在 HPM6700/6400 系列 MCU 上带有音频功能。支持四路 I2S 功能、一路 DAO 播音功能、一路 PDM 数字麦克风能和 VAD 功能。I2S 支持时分复用，最多支持 TDM16。I2S 模块支持四种模式：飞利浦标准、MSB 对齐模式、LSB 对齐模式和 DSP 模式。

本文以 HPM6750EVK2 开发板为例介绍 I2S 的 DMA 链式收发数据功能。接收到数据为麦克风采集的信号经过 CODEC 到 MCU；I2S 发送的数据经过 DAO 播放，外接喇叭即可播出声音。

2.整体方案

图 2-1 所示为 I2S 收发数据整体框图。主要包括两部分：麦克风采集数据并通过 CODEC 经 I2S 传输至 MCU、MCU 通过 I2S 将 1KHz 正弦波信号传输至 DAO，然后 经过 SPK 播放。

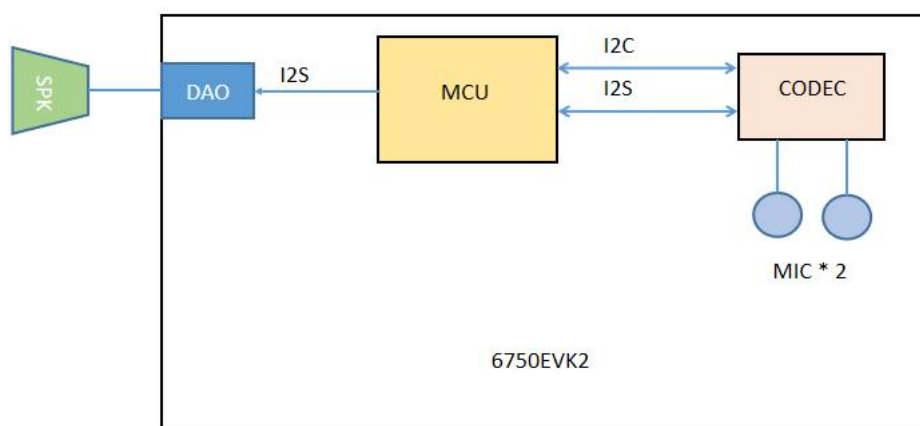


图 2-1 I2S 收发数据整体框图

图 2-2 所示为 I2S DMA 收发数据的软件框架。对于数据接收端，采用乒乓 buff 的方式存储 I2S 的 RX 数据，rx_buff 长度是每帧处理数据长度的 2 倍。

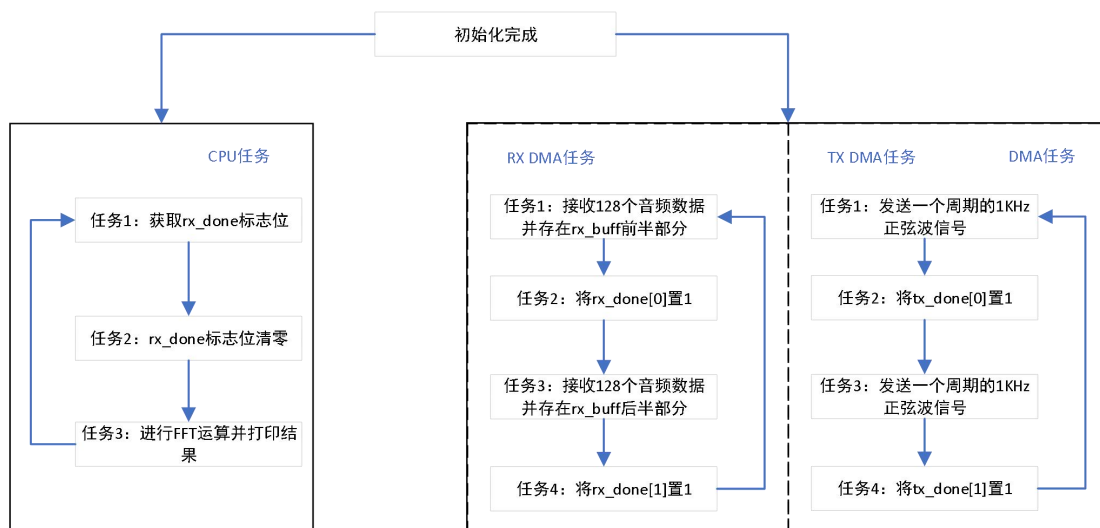


图 2-2 I2S DMA 收发数据软件框架

当 rx_buff 前半数据接收完成时，会给出完成标志，此时 CPU 可以处理前一

半数据，后一半数据的处理流程和前一半相同。CPU 只需要检查数据完成的标志并对数据处理，不需要参与数据的收发过程，大大降低了 CPU 的负荷。

例程中 rx_buff 和 rx_done 均放在 fast RAM 区，CPU 可以直接读取数据，减少读取数据消耗的时间。

I2S 发送功能是循环发送 1KHz 的正弦波信号到 DAO，通过 DMA 链发送，两个 DMA 任务发送的数据相同。

3.I2S DMA 接收功能

3.1 DMA 链式传输

HPM6000 系列的 DMA 支持链式连接多个任务，该方案利用这一特点，将 ADC 采样和 DMA 传输分为四个子任务，在 DMA 中断中自动完成任务的循环执行。这种方案不使用 CPU 的中断，降低 CPU 的负荷。

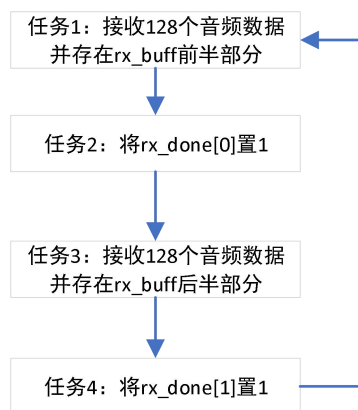


图 3-1 DMA 链接收数据流程

四个任务分别为：I2S 接收的数据（双声道，下同）存放至 rx_buff 前 256 个位置、rx_done[0]置 1、I2S 接收到数据存放至 rx_buff 后 256 个位置、rx_done[1]置 1。各任务通过 descriptors_rx 数组连接起来，顺序循环执行。

例程中每次 FFT 运算使用 128 个点，由于接收到是双声道数据，rx_buff 的长度是 $128 * 4 = 512$ 。DMA 链式传输时，首先将 I2S 接收数据存在前 256 个点，此时 rx_done[0] 置 1；接着 DMA 将接收到的数据存放在 rx_buff 的后 256 个点。传输完成后将 rx_done[1] 置 1，完成一个循环，然后进入下一个循环，如此往复，实现了 DMA 自动传输 I2S 数据的功能。

3.2 CPU 处理数据

DMA 负责将 I2S 接收到数据存放至 rx_buff 并在传输完成后给出标志位。CPU 的工作就是检测这些标志位，当标志位为 1 时将标志位清零并对数据进行处理。例程中 CPU 检测到 rx_done[0] 为 1 时，将 rx_done[0] 置 0。然后分别取左、右声道的数据，然后对右声道数据进行 FFT 运算，每计算 20 次会打印一次计算结果。当对前半部分的数据进行运算时，I2S 接收到数据将存在 rx_buff 的后半部分，数据存满时，rx_done[1] 置 1，例程中只是将 rx_done[1] 进行了清零，并没有对数据进行处理，用户可以根据需要对数据处理。

注意：每帧数据的算法处理时间应该小于数据接收到时间。以例程中的每帧处理单声道的 128 个点为例说明，因为采样率是 16000Hz，I2S 接收 128 个双声道数据的时间为 $128/16000 = 8\text{ms}$ ；因此当 rx_done[0] 或 rx_done[1] 为 1 时，每次处理数据的运行时间要小于 8ms，否则会出现原来的数据还未处理完，新数据又到了，造成数据被污染。建议用户测试下每帧的数据处理时间。

4. I2S DMA 发送功能

I2S 发送功能是将固定数据通过 I2S 将数据发送给 DAO。例程中发送的数

据是 1KHz 的正弦波信号。与 I2S 接收数据类似，发送数据也是通过 DMA 链的方式。

I2S 发送数据也是分为四个任务，四个任务分别为：I2S 发送双声道的 1KHz 正弦波信号、tx_done[0]置 1、I2S 接收到数据存放至 rx_buff 后 256 个位置、tx_done[1]置 1。各任务通过 descriptors_tx 数组连接起来，顺序循环执行。

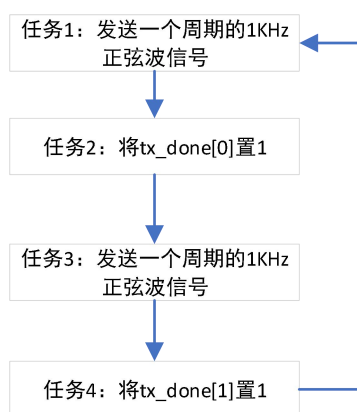


图 3-2 DMA 链发送数据流程

在 tx_done 标志位置 1 时，例程中并没有做任何处理，用户可以根据实际需要执行相应的处理，与处理接收到数据一样，需要注意 CPU 的负载情况。由于 I2S 的接收和发送是并行的，要求接收和发送总的处理时间要小于每帧接收时间。

5. 测试验证

本章将上文介绍的 I2S 通过 DMA 进行收发数据的方案在 HPM6750EVK2 开发板上进行验证。

5.1 实验准备

测试该方案需要一块功能正常的 HPM6750EVK2 开发板和一个可以连接到 DAO 到喇叭。

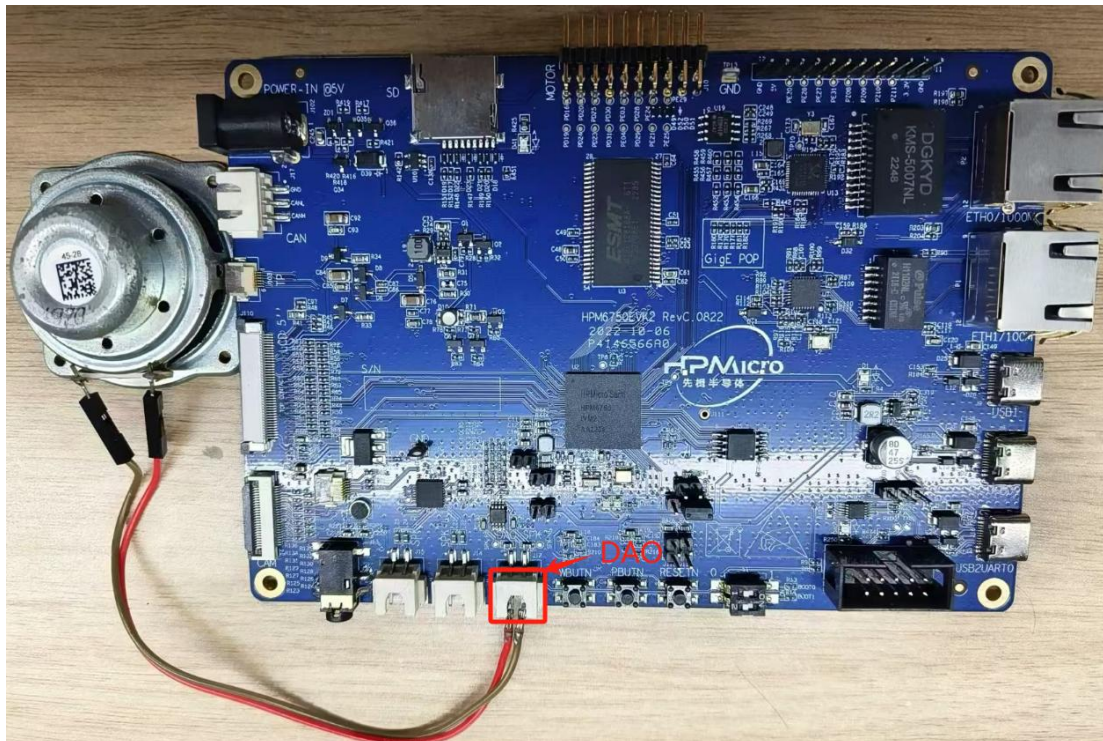


图 5-1 HPM6750EVK2 的 DAO 与喇叭接线

5.2 测试验证

步骤一：首先生成工程，方法如下图。将生成的工程打开并烧录到 HPM6750EVK2 开发板。

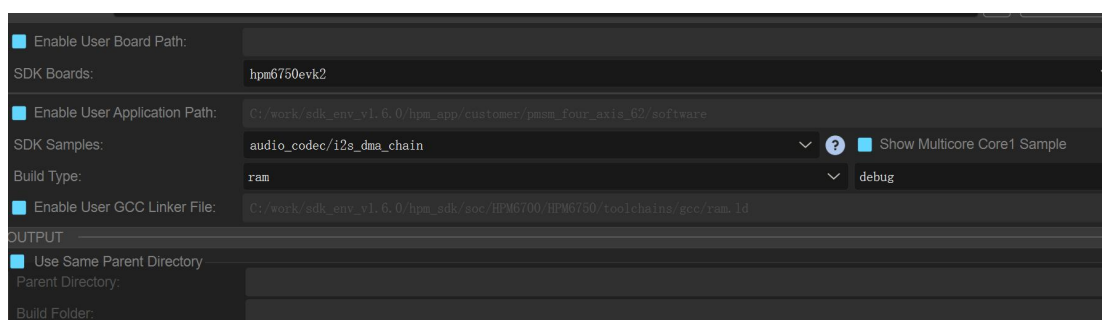


图 5-2 例程工程生成方法

步骤二：打开串口工具并与 MCU 连接，先不要将喇叭连接到 DAO 上，运行程序。此时会看到串口日志上会打印 FFT 的计算结果如图 5-2 所示，每计算

20 次会打印一次，每次打印四个频点的结果，其中第一个值表示 1KHz 频点处的能量值。由于 FFT 能量值偏大，打印的结果是经过移位处理的，从图中可以看出，没有外接喇叭播放 1KHz 正弦波信号时，四个频点处的能量值相差不大且相对较小。

注意：该测试结果是在相对安静的环境下进行的，如果有环境噪声或者播放音乐等情况，打印的结果会变化。

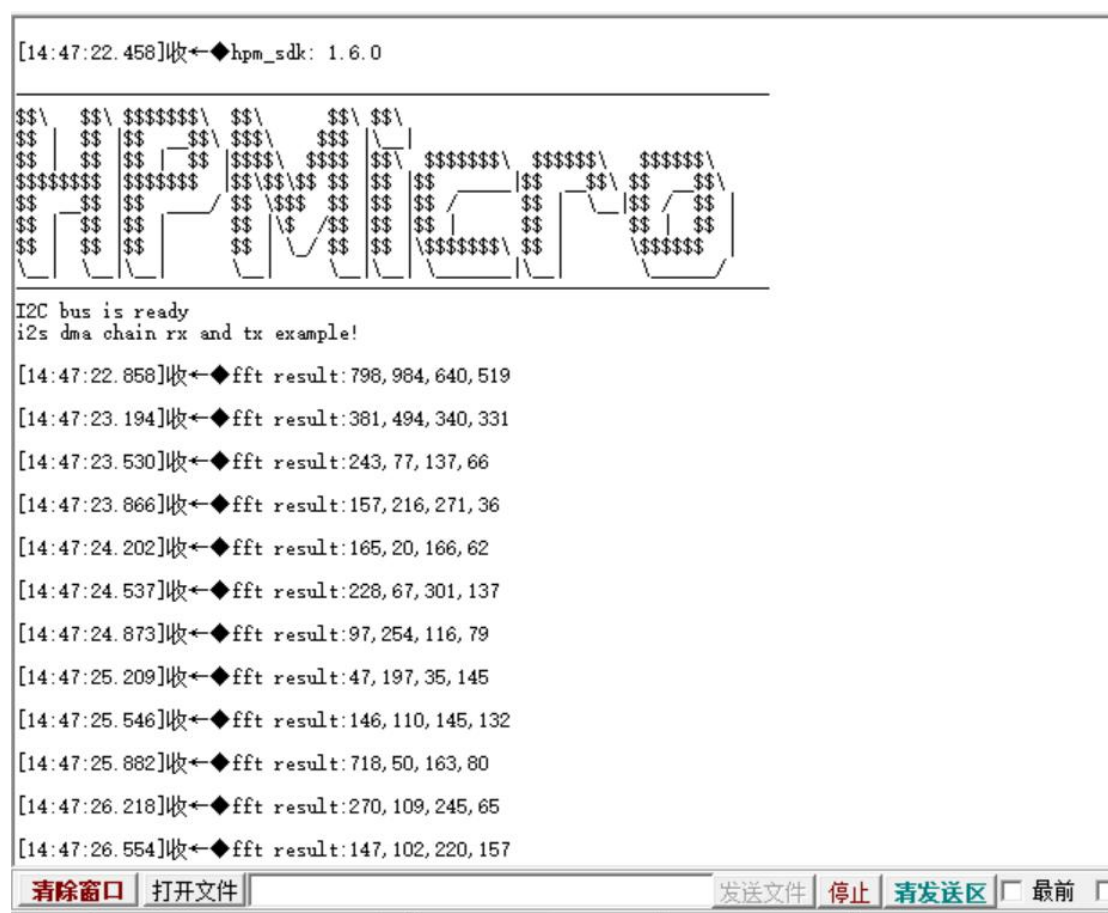


图 5-3 DAO 没有外接喇叭时 FFT 计算结果

步骤三：将 DAO 接上喇叭，会听到 1KHz 的正弦波声音，此时串口日志如图 5-4 所示。可以看出在 1KHz 频点处能量值达到最大（由于例程中对 FFT 结果做了处理，最大值为 16383）。当拔掉喇叭后，测试结果又会变小。

说明：例程中把 FFT 计算结果从 float 型强转成了 uint32 型，仅为测试使

用，用户实际使用时可直接使用原始结果。

```
[14:47:36.298]收←◆fft result:213,236,185,99
[14:47:36.635]收←◆fft result:93,146,88,120
[14:47:36.970]收←◆fft result:249,66,106,485
[14:47:37.306]收←◆fft result:122,90,210,167
[14:47:37.642]收←◆fft result:57,128,136,55
[14:47:37.978]收←◆fft result:16383,1027,378,662
[14:47:38.314]收←◆fft result:16383,283,403,412
[14:47:38.650]收←◆fft result:16383,350,160,169
[14:47:38.987]收←◆fft result:16383,267,273,367
[14:47:39.323]收←◆fft result:16383,184,62,169
[14:47:39.658]收←◆fft result:15446,802,391,483
[14:47:39.994]收←◆fft result:16383,258,121,157
[14:47:40.330]收←◆fft result:606,285,86,84
[14:47:40.666]收←◆fft result:466,184,128,234
[14:47:41.003]收←◆fft result:294,271,247,229
[14:47:41.338]收←◆fft result:527,189,113,199
[14:47:41.674]收←◆fft result:557,20,229,225
[14:47:42.011]收←◆fft result:406,272,41,168
[14:47:42.347]收←◆fft result:134,156,230,134
[14:47:42.682]收←◆fft result:46,80,115,81
```

图 5-4 DAO 外接喇叭后测试结果

如前文介绍，处理每帧数据的时间应该小于数据接收到时间。如果处理数据的时间大于接收数据的时间，串口会打印 rx_buff full!的信息，当出现此信息，需要先测一下处理数据所用的时间，使用优化算法、增加帧长度等手段解

决问题。

6.总结

本文主要介绍了通过 DMA 进行 I2S 数据收发功能的方案。数据的收发均是通过 DMA 链式任务自动完成，不需要 CPU 参与。CPU 只需要检测传输完成的标志位并对数据进行处理。在 HPM6750EVK2 上对方案进行测试验证，验证结果表明该方案可正常运行。

该方案的优点主要体现在以下几个方面：

- 1、使用 DMA 链式传输，自动完成 I2S 接收数据到 rx_buff 以及 I2S 发送正弦波信号到 DAO，不需要 CPU 参与。CPU 只负责处理数据；
- 2、没有使用中断，降低 CPU 的负荷；
- 3、I2S 接收到数据保存在 fast RAM 中，减小读取数据的延时时间。