# 第五章 系统测试与分析

## 5.1系统测试指标

解码速度：MP3解码一帧，即执行函数

***MP3Decode(HMP3Decoder, unsigned char \*\*, int \*, char \*, int);***

所需要耗费的时间。

下载速度：开始网络音乐下载后，音乐文件下载的平均速度。

## 5.2 测试环境

### 5.2.1验证开发平台

运算内核：EMSK2.3开发板平台 Arcem7d，内核频率25MHz

网络环境：基于联通4G网络的手机Wifi热点

### 5.2.2测试方案

解码速度测试：

分别在单独开启音乐解码任务和同时开启音乐解码任务和网络下载任务的情况下测试。

再分别测试四种优化级别下完成MP3解码函数的时间。

下载速度测试：

在音乐解码任务和网络下载任务同时启动的情况下，使用最高串口波特率：3125000与ESP8266进行通讯。在启动音乐文件下载后，计算单位时间内（1秒），网络数据的平均到达量。

## 5.3测试结果

### 5.3.1功能测试

进行完全的优化后，音乐解码播放，网络下载，GUI操作三个任务可以很好地配合，互不干扰地运行。

音乐可以流畅地播放，没有失真，噪声。网络下载可以顺利进行，在本首歌播放完成之前，下一首歌可以完成下载，进入准备播放状态。GUI控制流畅，操作没有延迟。

### 5.3.2指标测试

解码速度测试：

|  |  |  |
| --- | --- | --- |
| 优化情况 | 单任务状态解码用时 | 多任务状态解码用时 |
| 无优化 | 32ms | 任务无法调度 |
| 内联汇编封装DSP指令 | 19ms | 任务无法调度 |
| 内联汇编封装核心代码（不使用CCM） | 18ms | 26ms |
| 内联汇编封装核心代码（使用CCM） | 14ms | 18ms |

网络速度测试：

|  |  |  |  |
| --- | --- | --- | --- |
| 次数 | 最低值 | 最高值 | 平均值 |
| 1 | 5kB/s | 70kB/s | 约50kB/s |
| 2 | 10kB/s | 83kB/s | 约70kB/s |
| 3 | 30kB/s | 105kB/s | 约80kB/s |
| 4 | 17kB/s | 95kB/s | 约60kB/s |
| 5 | 19kB/s | 97kB/s | 约60kB/s |

## 5.3结果分析

5.3.1解码速度分析

本设计中使用的是EMSK2.3 arcem7d内核。其内核频率只有25MHz。而本设计中所使用的解码库，为了保证解码的精度并不受到损失，使用了大量32位、乘法，甚至将32位的参数转换为64位进行运算。因此，在不使用任何优化的情况下，解码运算耗时较长，情况不理想。

在使用了DSP指令后，大量运算可以使用DSP加速，减少运算周期，同时利用部分DSP指令的特性，可以在不损失运算精度的情况下避免64位运算。因此计算效率得到大幅度提升。

将核心运算部分使用内联函数封装，最大程度地利用内核寄存器，可以有效减小存储器读写的频率。充分发挥流水线的优势。

由于网络接收是使用中断接收数据到缓冲区，该缓冲区较大，位于DDR2中。因此网速较快时，中断触发也较频繁，将会影响解码运算的资源，同时，两个任务都需要读写内存，读写速度将大幅度限制流水线性能的发挥。

在使用了CCM之后，将所有代码和解码所需要用到的核心数据分别放到CCM中，取指令和取数据所需要的时间大幅度减少，流水线优势得到发挥。同时网络下载任务主要写DDR2，解码任务主要读写DCCM，两者之间的干扰大幅度减小，因此性能得到了很大提升。

5.3.1网络下载速度分析：

在本设计中使用了EMSK开发板可以使用的最高波特率3125000与ESP8266通讯，因此理论最高网络下载速度可以达到305kB/s。但实际最高下载速度只有105kB/s。据此可知，下载速度受限于外网提供的带宽，测试没有达到硬件可以承受的最高带宽。

# 第六章 总结展望

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# 参考文献

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*