Tpbulk.c 和 Tpbulk.h 中为 Mass Storage 类协议利用 Bulk 传输实现类协议请求命令。

为了读/写 U 盘中的基于 FAT 架构的文件，就需要在 Mass Storage 类的基础上，在主机系统中建立一个精简的 FAT 文件系统。这个文件系统是软件层次的程序设计，与 USB 及底层硬件无关，它利用 Mass Storage 类提供的各种类请求命令来进行文件的传输。FAT 文件系统将 U 盘的结构进行系统化，按照特定的规定以文件的形式来管理 U 盘中的数据。

通过 Mass Storage 类协议，USB 主机可以对 U 盘内特定的物理位置的数据进行读/写，这些数据无直接联系。而有了 FAT 文件系统，这些散乱在 U 盘内的数据就被软件化地组织起来，以一个统一的文件形式，被主机管理。

5.2.1 脱机工作固件流程

脱机工作固件流程如图 5-9 所示。

SL811 芯片接入 U 盘的启动过程是:①在系统初始化阶段，单片机已对 SL811 进行主模式下的初始化，接着，SL811 通过监测数据线 D+和 D-上的电平变化来检测是否有 U 盘插入;②有从设备插入,单片机对 SL811 进行 RESET，然后检测所插入设备是全速设备还是低速设备，并配置 SL811 相应的寄存器；③接着进入枚举阶段，单片通过 SL811 控制端点 0 读取该 USB 从设备的描述符，得到从设备的包括端点在内的所有属性及其他相关信息，并且给插入的 USB 设备分配地址；④由于插入的是 U 盘，所以要再次进行枚举，这次枚举是将设备识别为大规墨存储设备，并且初始化 FAT16 文件系统，这样就可对 U 盘进行磁盘操作，主要完成数据接入。

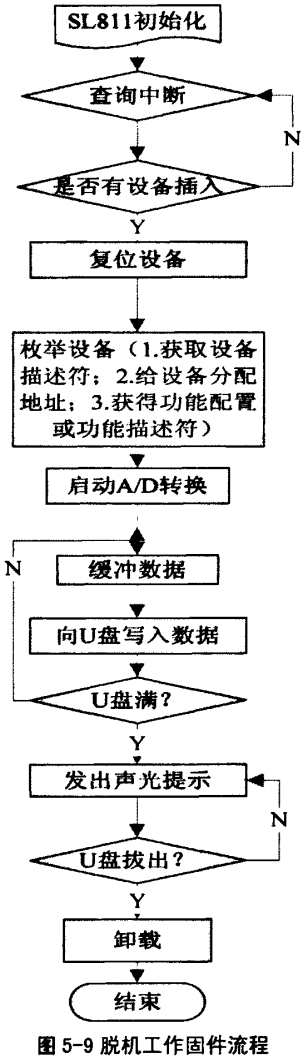

图 5-9 脱机工作固件流程

单片机在完成 SL811 接入 U 盘的启动过程之后，按预先设定的程序控制数据采集与转换；然后缓冲数据，将数据传输到 U 盘中进行存储，当 U 盘满并溢出时，发出声光提示并等待人工决策；系统能检测 U 盘的接入和拔出，U 盘拔出后系统将 U 盘自动卸载。

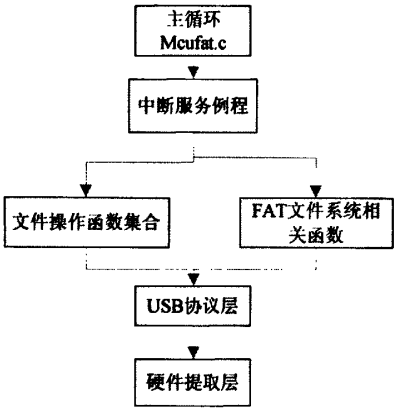

图 5-10 脱机工作固件结构

5.2.2 脱机工作固件编程的实现

脱机工作固件结构如图 5-10 所示。

1. 脱机工作固件主程序—Mcufat.c

通过监控 bBoardFlags.bits.bTimer、bBoardFlags.bits.SLAVE_FOUND、bBoardFlags.bits.bMassDevice、bBoardFlags.bits.BDATA_BUF、bBoardFlags.bits.bDataInDone、bBoardFlags.bits.bUPanFULL、bBoardFlags.bits.SLAVE_REMOVED 全局变量的变化，进入相应的处理函数进行处理。

在系统初始化阶段，单片机已对 SL811 进行主模式下的初始化，bBoardFlags.bits.SLAVE_FOUND=1 时说明有 U 盘插入，则进入相应的处理函数复位设备并枚举设备；bBoardFlags.bits.bMassDeviee=1 说明将 U 盘识别为大容量存储设备；然后启动 A/D 变换子程序； bBoardFlags.bits.BDATA_BUF =1 将采集到的数据进行缓存；bBoardFlags.bits.bDataInDone=1 时将数据写入 U 盘；bBoardFlags.bits.bUPanFULL=1 时，表示 U 盘已满，发出声光提示；bBoardFlags.bits.SLAVE_REMOVED =1 表示 U 盘已拔出，执行卸载操作。

2. 中断—Isrfat.c

主要是 Timer0 中断及 SL811 中断的相关处理操作。

Timer0 中断:voidISR_Timer0(void)inierrupt1。Timer0 被设置为 16 位计数器，自动计时。每当计数器溢出时，触发本中断。主要目的是为了每隔一定时间查询一次 USB 设备状态。

SL811 硬件中断:voidISR_INT0(void)interrupt0。当有 USB 设备接入时，就可以触发本中断。进入中断后要将中断清除，以便接收新的中断。

3. 文件操作函数集—Tpbulk.c 和 Tpbulk.h

实现了 USB Mass Storage 类协议的基本请求命令。