

# 1 前言

随着计算机技术的快速发展，USB（Universal Serial Bus）存储设备的使用已经非常普遍，USB 用于将适用 USB 的外围设备(device)连接到主机(host)，实现二者之间数据传输的外部总线结构；是一种快速、灵活的总线接口<sup>[5]</sup>。它最大的特点是易于使用，主要是用在中速和低速的外设。随着 USB 规范的完善和成熟，USB 外设的种类不断丰富，应用领域也不断扩大。在传统的应用中，主要是 PC 扮演着主机的角色。根据 USB 的规范，可以看到在 USB 的拓扑结构中居于核心地位的是主机，每一次的数据传输都必须由主机发起和控制。但是随着单片机产品应用领域的日益增长，USB 外设的应用范围也随之扩大，为此在单片机系统中实现对 USB 外设控制也变得日益迫切。因此在一些需要转存数据的设备，仪器上使用 USB 移动存储设备接口的芯片便相继产生了，CH375 就是其中之一，它是一个 USB 总线的通用接口芯片，支持 HOST 主机方式和 SLAVE 设备方式<sup>[4]</sup>。

如今的 USB 就象当日的 R232，最终发展必是业界的主流外设接口技术。USB 有着其它接口不可替代的优势<sup>[12]</sup>。

随着 USB 技术的发展，计算机的移动存储介质普遍采用 U 盘或移动硬盘。

如今，USB 技术已经越来越普及和成熟，低成本、高稳定性、较高的数据传输速率和即插即用的方便性，使其备受硬件厂商的青睐。随着数据采集和单片机用户对移动存储的需求越来越大，具有 USB 接口的存储设备以其优异的性价比和灵活性常用来进行数据的存储和交换，所以在单片机系统中实现对优盘或移动硬盘的直接读写是非常有价值的<sup>[6]</sup>。

介绍了一种 USB 总线的通用接口芯片 CH375，并在此基础上提出了一种外部单片机读写 U 盘的基本方法及其硬件连接方法。单片机只要在原硬件系统中增加 1 个 CH375 芯片就可以直接调用 CH375 提供的子程序库来直接读取 U 盘中的数据，从而实现了普通单片机与 U 盘的通讯、方法简单、便于操作、综合成本比较低，具有较大的推广应用价值<sup>[10]</sup>。

## 2 总体方案设计

### 2.1 电路原理图

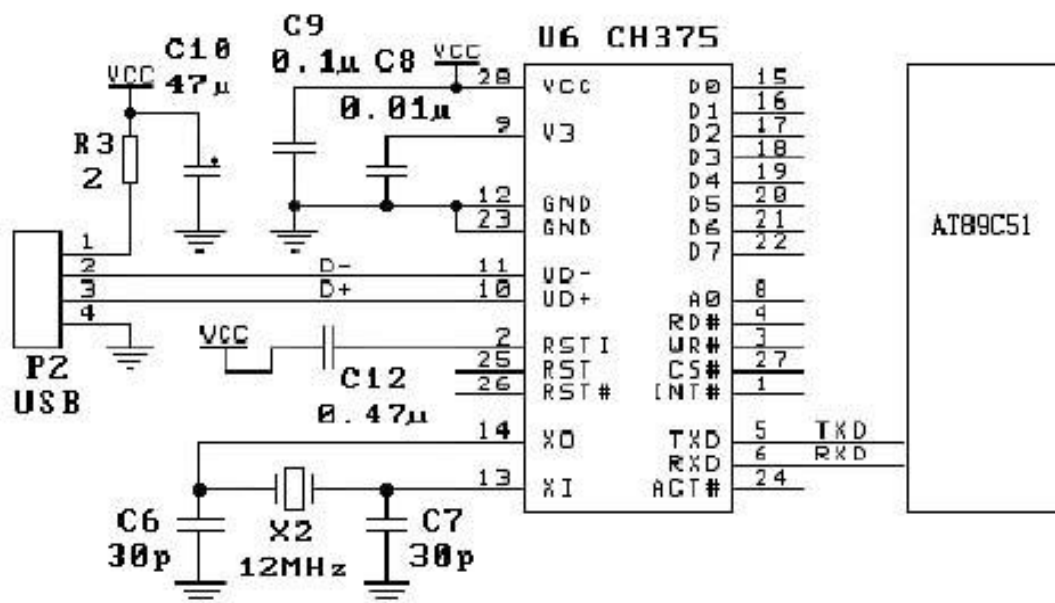


图1 MCS-51 单片机读写 U 盘的电路原理图

MCS-51 单片机读写 U 盘的电路原理图，CH375 芯片的 TXD 引脚接高电平，工作于串口方式。在串口方式下，CH375 只需要与单片机/DSP/MCU 连接 3 个信号线，TXD 引脚、RXD 引脚以及 INT#引脚，其他引脚都可以悬空。除了连接线较少之外，其他外围电路与并口方式基本相同。由于 INT#引脚和 TXD 引脚在 CH375 复位期间只能提供微弱的高电平输出电流，在进行较远距离的连接时，为了避免 INT#或者 TXD 在 CH375 复位期间受到干扰而导致单片机误操作，可以在 INT#引脚或者 TXD 引脚上加阻值为  $1\sim 5k\Omega$  的上拉电阻，以维持较稳定的高电平。在 CH375 芯片复位完成后，INT#引脚和 TXD 引脚将能够提供 5mA 的高电平输出电流或者 5mA 的低电平吸入电流。

### 2.2 硬件结构

CH375 工作于并口方式，与普通的 MCS-51 单片机的连接如图所示。CH375 的 TXD 引脚通过  $1k\Omega$  左右的下拉电阻接地或者直接接地，从而使 CH375 工作于并口方式。

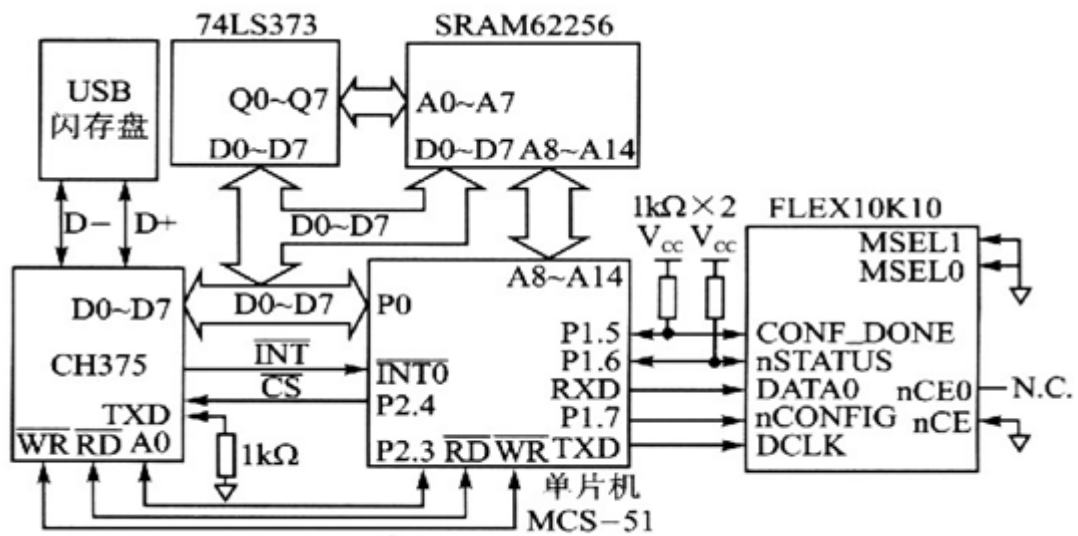


图 2 硬件结构图

## 2.3 软件设计

①进入 USB 主机模式，要读/写 U 盘，必须先设置 CH375 使其工作于 USB 主机方式。②将 U 盘当作存储器进行读 / 写，方法与读 / 写闪存差不多，操作简单且速度快。单片机读 / 写 U 盘文件程序分层如图所示。

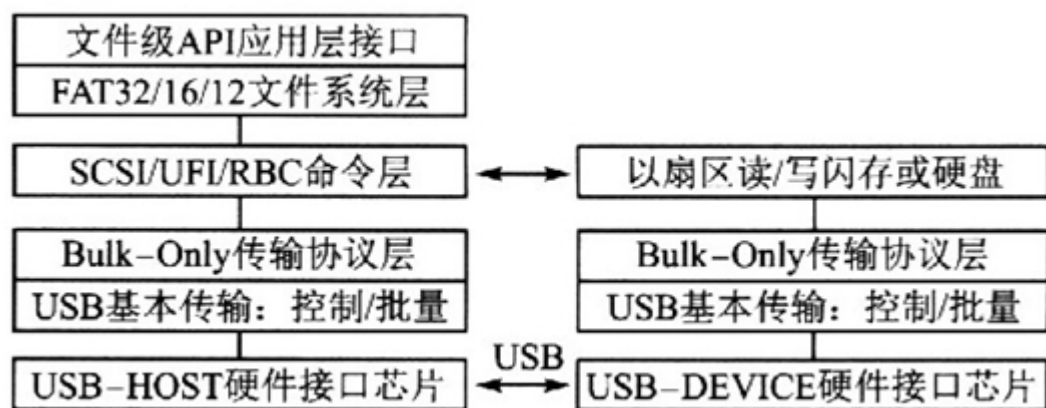


图3 单片机读/写U盘文件程序分层图

## 2.4 软件设计流程图

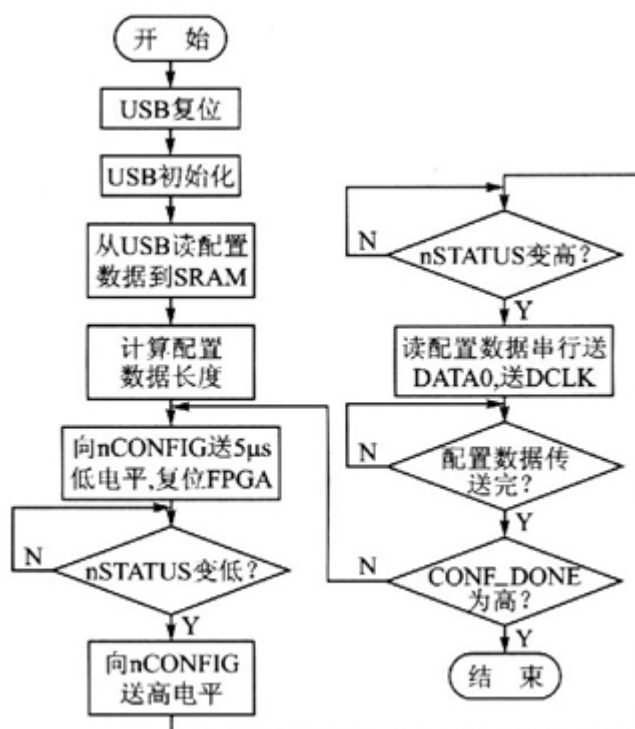


图 4 软件设计流程图

## 3 51 单片机的结构及组成

### 3.1 51 单片机的内部结构

1、中央处理单元（8 位）

数据处理、测试位，置位，复位 位操作

2、只读存储器（4KB 或 8KB）

永久性存储应用程序，掩模 ROM、EPROM、EEPROM

3、随机存取内存（128B、128B SFR）

在程序运行时存储工作变量和资料

4、并行输入/输出口（I / O）（32 条）

作系统总线、扩展外存、I / O 接口芯片

5、串行输入/输出口（2 条）

串行通信、扩展 I / O 接口芯片

6、定时/计数器（16 位、加 1 计数）

计满溢出、中断标志置位、向 CPU 提出中断请求，与 CPU 之间独立工作

7、时钟电路

内振、外振。

8、中断系统

五源中断、2 级优先。

### 3.2 51 单片机的结构特点

MCS-51 系列单片机为哈佛结构（而非普林斯顿结构）

1) 内 ROM: 4KB

2) 内 RAM: 128B

3) 外 ROM: 64KB

4) 外 RAM: 64KB

5) I / O 线: 32 根（4 埠，每埠 8 根）

6) 定时/计数器: 2 个 16 位可编程定时/计数器

- 7) 串行口：全双工，2 根
- 8) 寄存器区：工作寄存器区、在内 128B RAM 中，分 4 个区，
- 9) 中断源：5 源中断，2 级优先
- 10) 堆栈：最深 128B
- 11) 布尔处理机：位处理机，某位单独处理
- 12) 指令系统：五大类，111 条

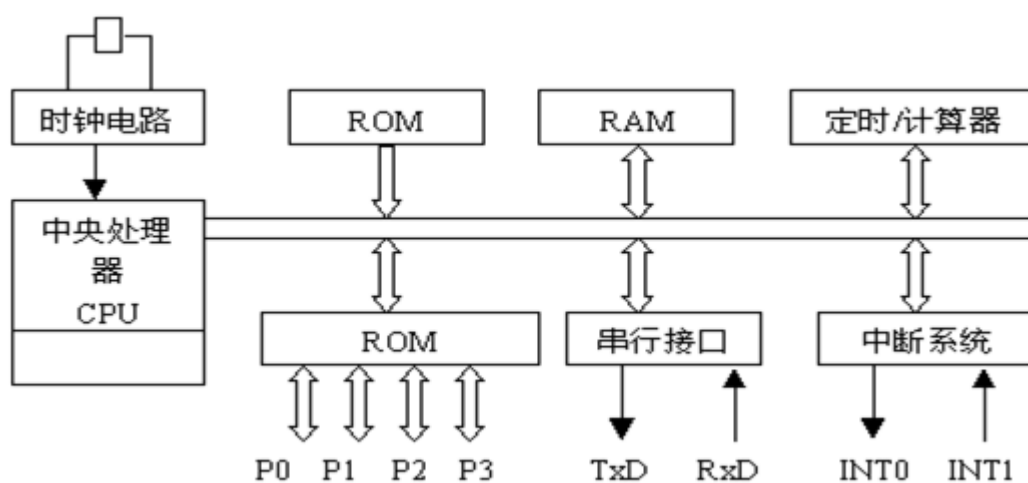


图 5 MCS-51 单片机结构框图

### 3.3 MCS-51 的 CPU 的组成

MCS-51 的 CPU 能处理 8 位二进制数或代码。CPU 是单片机的主要核心部件，在 CPU 里面包含了运算器、控制器以及若干寄存器等部件组成。

(1) 内部数据存储器 (RAM)：

MCS-51 单片机芯片共有 256 个 RAM 单元，其中后 128 单元被专用寄存器占用，能作为寄存器供用户使用的只是前 128 单元，用于存放可读写的数。因此通常所说的内部数据存储器就是指前 128 单元，简称内部 RAM。地址范围为 00H~FFH (256B)。是一个多用多功能数据存储器，有数据存储、通用工作寄存器、堆栈、位地址等空间。

(2) 内部程序存储器 (ROM)：

MCS-51 内部有 4KB/8KB 字节的 ROM (51 系列为 4KB，51 系列为 8KB)，用于存放程序、原始数据或表格。因此称之为程序存储器，简称内部 RAM。地址范围为 0000H~FFFFH (64KB)。

### (3)定时器/计数器

51 系列共有 2 个 16 位的定时器/计数器（52 系列共有 3 个 16 位的定时器/计数器），以实现定时或计数功能，并以其定时或计数结果对计算机进行控制。定时时靠内部分频时钟频率计数实现，做计数器时，对 P3.4（T0）或 P3.5（T1）端口的低电平脉冲计数。

### (4)并行 I/O 口

MCS-51 共有 4 个 8 位的 I/O 口（P0、P1、P2、P3）以实现数据的输入输出。

### (5)串行口

MCS-51 有一个可编程的全双工的串行口，以实现单片机和其它设备之间的串行数据传送。该串行口功能较强，既可作为全双工异步通信收发器使用，也可作为移位器使用。RXD（P3.0）脚为接收端口，TXD（P3.1）脚为发送端口。

### (6)中断控制系统

MCS-51 单片机的中断功能较强，以满足不同控制应用的需要。51 系列有 5 个中断源（52 系列有 6 个中断源），即外中断 2 个，定时中断 2 个，串行中断 1 个，全部中断分为高级和低级共二个优先级别。

### (7)定时与控制部件

MCS-51 单片机内部有一个高增益的反相放大器，基输入端为 XTAL1 输出端为 XTAL2。MCS-51 芯片的内部有时钟电路，但石英晶体和微调电容需外接。时钟电路为单片机产生时钟脉冲序列。

## 4 USB 总线接口芯片 CH375

### 4.1 概述<sup>[1]</sup>

CH375 是一个USB总线的通用接口芯片，支持USB-HOST主机方式和USB-DEVICE/SLAVE设备方式。在本地端，CH375具有8位数据总线和读、写、片选控制线以及中断输出，可以方便地挂接到单片机/DSP/MCU/MPU等控制器的系统总线上。在USB主机方式下，CH375还提供了串行通讯方式，通过串行输入、串行输出和中断输出与单片机/DSP/MCU/MPU 等相连接。

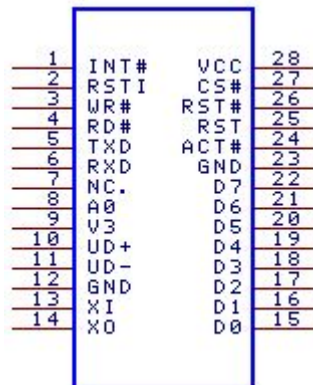
CH375的USB 主机方式支持常用的USB 全速设备，外部单片机可以通过CH375 按照相应的USB 协议与USB 设备通讯。CH375 还内置了处理Mass-Storage 海量存储设备的专用通讯协议的固件，外部单片机可以直接以扇区为基本单位读写常用的USB 存储设备（包括USB 硬盘/USB 闪存盘/U 盘）

### 4.2 特点

- 全速USB-HOST 主机接口，兼容USB V2.0，外围元器件只需要晶体和电容。
- 全速设备接口，完全兼容CH372 芯片，支持动态切换主机与设备方式。
- 主机端点输入和输出缓冲区各64 字节，支持常用的12Mbps 全速USB 设备。
- 支持USB 设备的控制传输、批量传输、中断传输。
- 自动检测USB 设备的连接和断开，提供设备连接和断开的事件通知。
- 内置控制传输的协议处理器，简化常用的控制传输。
- 内置固件处理海量存储设备的专用通讯协议，支持Bulk-Only传输协议和SCSI、UFI、RBC 或等效命令集的USB 存储设备（包括USB 硬盘/USB 闪存盘/U 盘）。
- 通过U 盘文件级子程序库实现单片机读写USB 存储设备中的文件。
- 并行接口包含8 位数据总线，4 线控制：读选通、写选通、片选输入、中断输出。
- 串行接口包含串行输入、串行输出、中断输出，支持通讯波特率动态调整。
- 支持5V 电源电压和3.3V 电源电压，CH375A 芯片还支持低功耗模式。
- 采用SOP-28 封装，可以提供SOP28 到DIP28 的转换板。



## 4. 3 封装



封装形式	塑体宽度		引脚间距		封装说明	订货型号
SOP-28	7.62mm	300mil	1.27mm	50mil	标准的 28 脚贴片	CH375S/A

## 4. 4 引脚

引脚号	引脚名称	类型	引脚说明
28	VCC	电源	正电源输入端，需要外接 0.1 $\mu$ F 电源退耦电容
12、23	GND	电源	公共接地端，需要连接 USB 总线的地线
9	V3	电源	在 3.3V 电源电压时连接 VCC 输入外部电源，在 5V 电源电压时外接容量为 0.01 $\mu$ F 退耦电容
13	XI	输入	晶体振荡的输入端，需要外接晶体及振荡电容
14	XO	输出	晶体振荡的反相输出端，需要外接晶体及振荡电容
10	UD+	双向三态	USB 总线的 D+ 数据线，内置可控的上拉电阻
11	UD-	双向三态	USB 总线的 D- 数据线
22~15	D7~D0	双向三态	8 位双向数据总线，内置上拉电阻
4	RD#	输入	读选通输入，低电平有效，内置上拉电阻
3	WR#	输入	写选通输入，低电平有效，内置上拉电阻
27	CS#	输入	片选控制输入，低电平有效，内置上拉电阻
1	INT#	输出	在复位完成后为中断请求输出，低电平有效
8	AO	输入	地址线输入，区分命令口与数据口，内置上拉电阻，当 AO=1 时可以写命令，当 AO=0 时可以读写数据
24	ACT#	输出	在内置固件的 USB 设备方式下是 USB 设备配置完成状态输出，低电平有效。对于 CH375A 芯片，在 USB 主机方式下是 USB 设备连接状态输出，低电平有效
5	TXD	输入输出	仅用于 USB 主机方式，设备方式只支持并口，在复位期间为输入引脚，内置上拉电阻，如果在复位期间输入低电平那么使能并口，否则使能串口，复位完成后为串行数据输出
6	RXD	输入	串行数据输入，内置上拉电阻
2	RSTI	输入	外部复位输入，高电平有效，内置下拉电阻
25	RST	输出	电源上电复位和外部复位输出，高电平有效
26	RST#	输出	电源上电复位和外部复位输出，低电平有效
7	NC.	空脚	空脚，必须悬空

## 4.5 功能说明

### 4.5.1 一般说明

CH375 芯片可以工作于USB-HOST 主机方式或者USB 设备方式。

CH375的USB 主机方式支持并行接口和串行接口。在USB主机方式下，CH375支持各种常用的USB 全速设备，外部单片机需要编写固件程序按照相应的USB协议与USB 设备通讯。但是对于USB 存储设备，CH375 内置了相关协议，通常情况下，外部单片机不需要编写固件程序，就可以直接通讯。

### 4.5.2 本地端的硬件

CH375 芯片在本地端提供了通用的被动并行接口和点对点的串行接口。

#### 4.5.2.1 并行接口

并口信号线包括：8 位双向数据总线D7~D0、读选通输入引脚RD#、写选通输入引脚WR#、片选输入引脚CS#、中断输出引脚INT#以及地址输入引脚A0。通过被动并行接口，CH375芯片可以很方便地挂接到各种8 位单片机、DSP、MCU 的系统总线上，并且可以与多个外围器件共存。

CH375芯片的RD#和WR#可以分别连接到单片机的读选通输出引脚和写选通输出引脚。CS#由地址译码电路驱动，用于当单片机具有多个外围器件时进行设备选择。INT#输出的中断请求是低电平有效，可以连接到单片机的中断输入引脚或者普通I/O 引脚，单片机可以使用中断方式或者查询方式获知中断请求。

当WR#为高电平并且CS#和RD#及A0 都为低电平时，CH375 中的数据通过D7~D0输出；当RD#为高电平并且CS#和WR#及A0 都为低电平时，D7~D0上的数据被写入CH375 芯片中；当RD#为高电平并且CS#和WR#都为低电平而A0 为高电平时，D7~D0 上的数据被作为命令码写入CH375 芯片中。

#### 4.5.2.2 串行接口

串行接口只能用于USB 主机方式，CH375 芯片的USB 设备方式不支持串口。

串口信号线包括：串行数据输入引脚RXD、串行数据输出引脚TXD、中断输出引脚INT#。

通过串行接口，CH375 可以用最少的连线与单片机、DSP、MCU 进行较远距离的点对点连接。

CH375芯片的RXD 和TXD可以分别连接到单片机的串行数据输出引脚和串行数据输入引脚。INT#输出的中断请求是低电平有效，用于通知单片机。

CH375 的串行数据格式是1个起始位、9个数据位、1个停止位，其中前8个数据位是一个字节数据，最后1个数据位是命令标志位。第9位为0时，前8位的数据被写入CH375芯片中，第9位为1时，前8位被作为命令码写入CH375芯片中。CH375的串行通讯波特率默认是9600bps，单片机可以随时通过SET\_BAUDRATE 命令选择合适的通讯波特率。

#### 4.5.2.3 其它

在CH375 芯片的复位期间，TXD引脚用于选择通讯接口。如果CH375 在复位期间检测到TXD 引脚为低电平则启用并行接口，否则启用串行接口。如果启用串行接口，那么复位完成后TXD 引脚将用于串行数据输出，并且CH375 芯片只能工作于USB 主机方式。

CH375芯片的ACT#引脚用于状态指示。在内置固件的USB设备方式下，当USB 设备尚未配置或者取消配置后，该引脚输出高电平；当USB 设备配置完成后，该引脚输出低电平。对于CH375A 芯片，在USB主机方式下，当USB 设备断开后，该引脚输出高电平；当USB设备连接后，该引脚输出低电平。CH375 的ACT#引脚可以外接串了限流电阻的发光二极管LED，用于指示相关的状态。

CH375 芯片的UD+和UD-引脚是USB 信号线，工作于USB 设备方式时，应该直接连接到USB 总线上；工作于USB 主机方式时，可以直接连接到USB设备。如果为了芯片安全而串接保险电阻或者电感，那么交直流等效串联电阻应该在 $5\Omega$ 之内。

CH375芯片内置了电源上电复位电路，一般情况下，不需要外部提供复位。RSTI 引脚用于从外部输入异步复位信号；当RSTI 引脚为高电平时，CH375芯片被复位；当RSTI 引脚恢复为低电平后，CH375会继续延时复位20ms 左右，然后进入正常工作状态。为了在电源上电期间可靠复位并且减少外部干扰，可以在RSTI引脚与VCC 之间跨接一个容量为0.47 $\mu$ F 左右的电容。RST引脚和RST#引脚是复位状态输出引脚，分别是高电平有效和低电平有效；当CH375电源上电复位或者被外部强制复位以及复位延时期间，RST 引脚和RST#引脚分别输出高电平和低电平；CH375复位完成后，RST 引脚和RST#引脚分别恢复到低电平和高电平。RST 和RST#引脚可以用于向外部单片机提供上电复位信号。

CH375芯片正常工作时需要外部为其提供12MHz 的时钟信号。一般情况下，时钟信号由CH375内置的反相器通过晶体稳频振荡产生。外围电路只需要在XI 和X0 引脚之间连接一个标称频率为12MHz的晶体，并且分别为XI 和X0 引脚对地连接一个高频振荡电容。如果从外部直接输入12MHz时钟信号，那么应该从XI 引脚输入，而X0 引脚悬空。

CH375芯片支持5V电源电压或者3.3V 电源电压。当使用5V 工作电压时，CH375 芯片的VCC引脚输入外部5V 电源，并且V3 引脚应该外接容量为0.01uF 左右的电源退耦电容。当使用3.3V 工作电压时，CH375 芯片的V3引脚应该与VCC引脚相连接，同时输入外部的3.3V 电源，并且与CH375 芯片相连接的其它电路的工作电压不能超过3.3V

#### 4.5.3 内部结构

CH375 芯片内部集成了PLL 倍频器、主从USB 接口SIE、数据缓冲区、被动并行接口、异步串行接口、命令解释器、控制传输的协议处理器、通用的固件程序等。

PLL 倍频器用于将外部输入的12MHz 时钟倍频到48MHz，作为USB 接口SIE 时钟。

主从USB 接口SIE是USB主机方式和USB设备方式的一体式SIE，用于完成物理的USB数据接收和发送，自动处理位跟踪和同步、NRZI编码和解码、位填充、并行数据与串行数据之间的转换、CRC数据校验、事务握手、出错重试、USB 总线状态检测等。

数据缓冲区用于缓冲USB 接口SIE 收发的数据。

被动并行接口用于与外部单片机/DSP/MCU 交换数据。

异步串行接口用于代替被动并行接口与外部单片机/DSP/MCU 交换数据。

命令解释器用于分析并执行外部单片机/DSP/MCU 提交的各种命令。

控制传输的协议处理器用于自动处理常用的控制传输的多个阶段，简化外部固件编程。

通用的固件程序包含两组：一组用于USB 设备方式，自动处理USB默认端点0 的各种标准事务等；另一组用于USB 主机方式，自动处理Mass-Storage 海量存储设备的专用通讯协议。

CH375 芯片内部具有7 个物理端点：

端点0 是默认端点，支持上传和下传，上传和下传缓冲区各是8 个字节；

端点1包括上传端点和下传端点，上传和下传缓冲区各是8个字节，上传端点的端点号是81H，下传端点的端点号是01H；

端点2 包括上传端点和下传端点，上传和下传缓冲区各是64 个字节，上传端点的端点号

是82H，下传端点的端点号是02H；

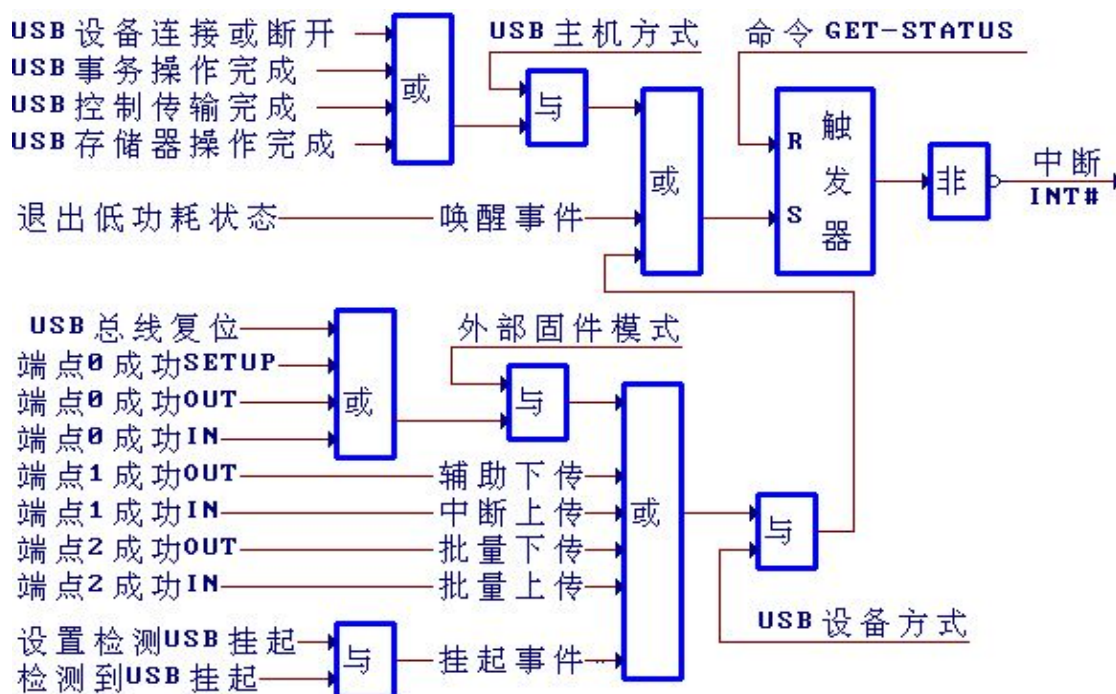
主机端点包括输出端点和输入端点，输出和输入缓冲区各是64 个字节，主机端点与端点2合用同一组缓冲区，主机端点的输出缓冲区就是端点2 的上传缓冲区，主机端点的输入缓冲区就是端点2 的下传缓冲区。

CH375 的端点0、1、2 只用于USB 设备方式，在USB 主机方式下只需要用到主机端点。

在USB 主机方式下，CH375 支持各种常用的USB 全速设备。USB 设备的端点号可以是0～15，两个方向最多支持31 个端点，USB 设备的包长度可以是0～64 字节。

内置固件可以处理Mass-Storage 海量存储设备的通讯协议，要求USB 存储设备支持Bulk-Only传输协议，支持SCSI、UFI、RBC或者等效的命令集，并且数据端点的最大包长度是64 字节，但是默认端点0 的最大包长度可以是8、16、32 或者64 字节。如果USB 存储设备不符合上述要求，则需要外部单片机通过控制传输以及ISSUE\_TOKEN 命令或者ISSUE\_TKN\_X 命令自行处理相关通讯协议。

下图为 CH375A 芯片内部的中断逻辑图。



#### 4.5.4 本地端的单片机软件

CH375芯片占用两个地址位，当A0 引脚为高电平时选择命令端口，可以写入命令；当A0 引脚为低电平时选择数据端口，可以读写数据。

单片机通过8 位并口对CH375 芯片进行读写，所有操作都是由一个命令码、若干个输入数据和若干个输出数据组成，部分命令不需要输入数据，部分命令没有输出数据。命令操作步骤如下：

- ①、 在A0=1 时向命令端口写入命令代码；
- ②、 如果该命令具有输入数据，则在A0=0 时依次写入输入数据，每次一个字节；
- ③、 如果该命令具有输出数据，则在A0=0 时依次读取输出数据，每次一个字节；
- ④、 命令完成，可以暂停或者转到①继续执行下一个命令。

CH375芯片专门用于处理USB 通讯，在检测到USB 总线的状态变化时或者命令执行完成后，CH375以中断方式通知单片机进行处理。

## 4.6 参数

### 4.6.1 绝对最大值（临界或者超过绝对最大值将可能导致芯片工作不正常甚至损坏）

名称	参数说明		最小值	最大值	单位
TA	工作时的环境温度	VCC=5V	-40	85	℃
		VCC=3.3V	-20	70	
TS	储存时的环境温度		-55	125	℃
VCC	电源电压（VCC 接电源，GND 接地）		-0.5	6.5	V
VIO	输入或者输出引脚上的电压		-0.5	VCC+0.5	V

### 4.6.2 电气参数（测试条件：TA=25℃，VCC=5V，不包括连接USB 总线的引脚）

（如果电源电压为3.3V，则表中所有电流参数需要乘以40%的系数）

名称	参数说明		最小值	典型值	最大值	单位
VCC	电源电压	V3 引脚不连 VCC 引脚	4.7	5	5.3	V
		V3 引脚连接 VCC 引脚	3.2	3.3	3.6	
ICC	工作时的总电源电流	VCC=5V		12	30	mA
		VCC=3.3V		6	15	
ISLP	低功耗状态的电源电流 I/O 引脚悬空/内部上拉	VCC=5V		0.25		mA
		VCC=3.3V		0.1		
VIL	低电平输入电压		-0.5		0.7	V
VIH	高电平输入电压		2.0		VCC+0.5	V
VOL	低电平输出电压（4mA 吸入电流）				0.5	V
VOH	高电平输出电压（4mA 输出电流）		VCC-0.5			V
IUP	内置上拉电阻的输入端的输入电流		40	80	160	uA
IDN	内置下拉电阻的输入端的输入电流		-80	-140	-240	uA
VR	电源上电复位的电压门限		2.4	2.7	3.0	V

注：ACT#引脚的低电平吸入电流为4mA，高电平输出电流为200uA。

在CH375 芯片复位期间INT#引脚和TXD 引脚只能提供80uA 的高电平输出电流。

#### 4.6.3 时序参数（测试条件：TA=25℃，VCC=5V 或者VCC=V3=3.3V，参考附图）

（RD 是指RD#信号有效并且CS#信号有效，RD#=CS#=0 执行读操作）

（WR 是指WR#信号有效并且CS#信号有效，WR#=CS#=0 执行写操作）

（适用于CH375A；如果是CH375S，那么TE0=TSX=TSC=TCC=2.5uS，TSD=TC=1.5uS）

名称	参数说明	最小值	典型值	最大值	单位
FCLK	USB 主机方式 XI 引脚的输入时钟的频率	11.995	12.00	12.005	MHz
TPR	电源上电的复位时间	18	20	40	mS
TRI	外部复位输入的有效信号宽度	100			nS
TRD	外部复位输入后的复位延时	18	20	40	mS
TE1	RESET_ALL 命令的执行时间		20	40	mS
TE2	SET_USB_MODE 命令的执行时间		10	20	uS
TE3	SET_ENDP?命令的执行时间		3	4	uS
TE4	SET_BAUDRATE 命令的执行时间	100		1000	uS
TE0	其余命令的执行时间			2	uS
TSX	命令码与命令码之间的间隔时间	2			uS
TSC	命令码与数据之间的间隔时间	2		100	uS
TSD	数据与数据之间的间隔时间	1		100	uS
TCC	命令写操作的周期	2			uS
TC	数据读操作或者写操作的周期	1		100	uS
TWW	有效的写选通脉冲 WR 的宽度	90		10000	nS
TRW	有效的读选通脉冲 RD 的宽度	90		10000	nS
TAS	RD 或 WR 前的地址输入建立时间	5			nS
TAH	RD 或 WR 后的地址输入保持时间	5			nS
TIS	写选通 WR 前的数据输入建立时间	0			nS
TIH	写选通 WR 后的数据输入保持时间	5			nS
TON	读选通 RD 有效到数据输出有效	0		30	nS
TOF	读选通 RD 无效到数据输出无效	0		20	nS
TINT	收到 GET_STATUS 命令 到 INT#引脚撤消中断请求		2	3	uS
TWAK	从低功耗状态退出的唤醒时间	2	3	5	mS

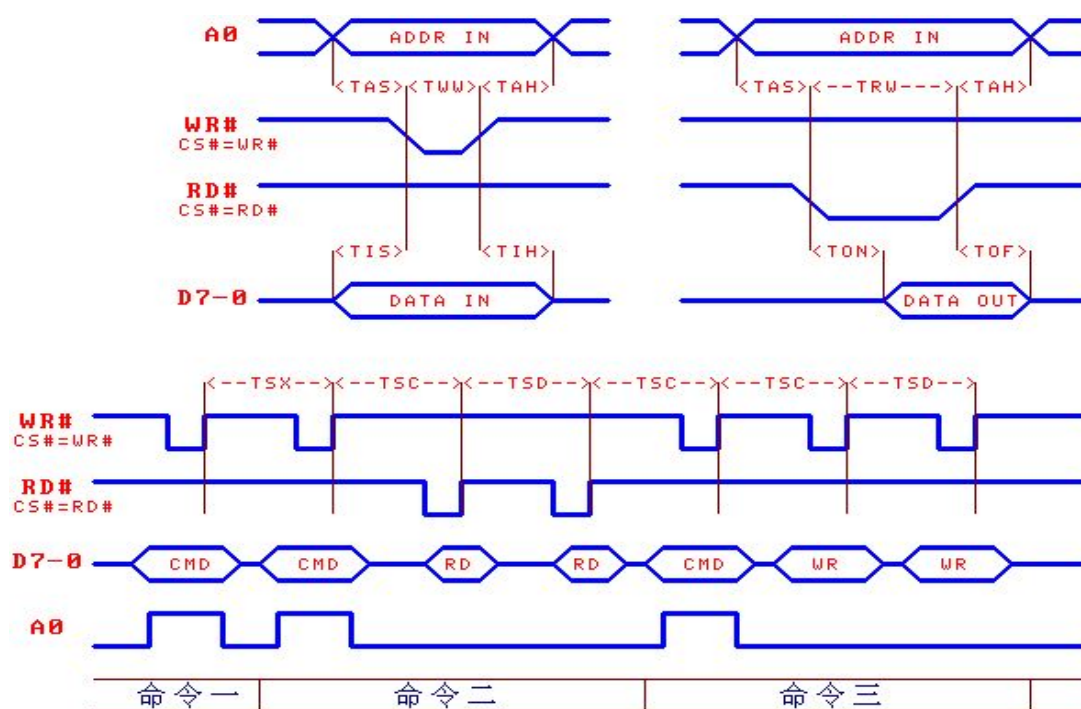


图 7



## 5 应用

### 5.1 并口方式（下图）

这是CH375与普通的MCS-51 单片机的连接电路。CH375 的TXD引脚通过1K $\Omega$ 左右的下拉电阻接地或者直接接地，从而使CH375 工作于并口方式。

USB 总线包括一对5V 电源线和一对数据信号线，通常，+5V 电源线是红色，接地线是黑色，D+信号线是绿色，D-信号线是白色。USB 插座P1 可以直接连接USB 设备，必要时可以在提供给USB 设备的+5V 电源线上串接具有有限流作用的快速电子开关，USB 电源电压必须是5V。

电容C3 用于CH375 内部电源节点退耦，C3 是容量为0.01 $\mu$ F 的独石或高频瓷片电容，如果对EMI 没有要求那么可以省掉C3。电容C4和C5用于外部电源退耦，C4 是容量为0.1 $\mu$ F 的独石或高频瓷片电容。晶体X1、电容C1 和C2 用于CH375 的时钟振荡电路。USB-HOST 主机方式要求时钟频率比较准确，X1 的频率是12MHz $\pm$ 0.4%，C1 和C2 是容量约为15pF 的独石或高频瓷片电容。

如果电源上电过程较慢并且电源断电后放电时间较长，那么CH375将不能可靠复位。可以在RSTI引脚与VCC 之间跨接一个容量为0.47 $\mu$ F 的电容C11，同时可以减少干扰。

在设计印刷线路板PCB 时，需要注意：退耦电容C3 和C4 尽量靠近CH375 的相连引脚；使D+和D-信号线贴近平行布线，尽量在两侧提供地线或者覆铜，减少来自外界的信号干扰；尽量缩短XI 和X0 引脚相关信号线的长度，为了减少高频时钟对外界的干扰，可以在相关元器件周边环境线或者覆铜。

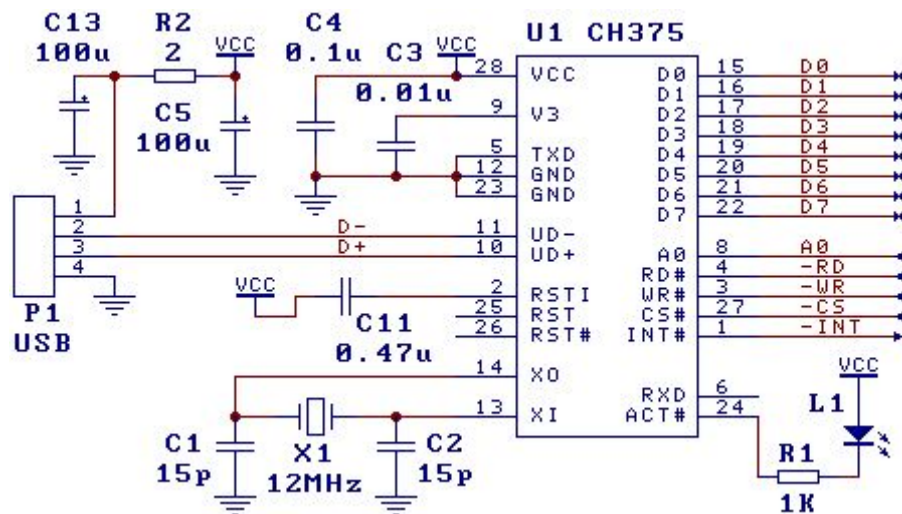


图 8 并口电路图



### 5.3 单片机读写U盘文件（USB 存储设备的文件级接口）

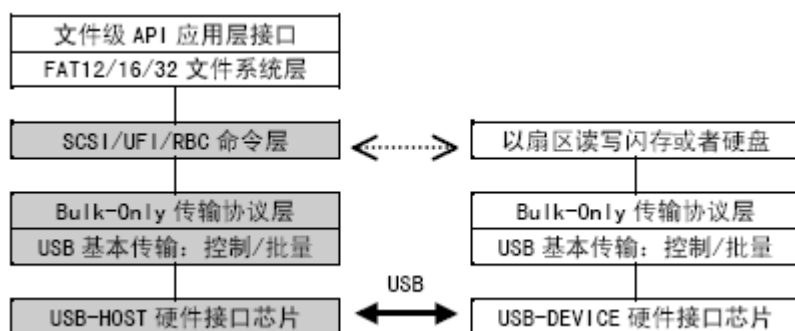


图 10 单片机读写 U 盘文件

一般情况下，单片机或嵌入式系统处理USB 存储设备的文件系统需要实现上图左边的4个层次，右边是USB 存储设备的内部结构层次。由于CH375 不仅是一个通用的USB-HOST 硬件接口芯片，还内置了相关的固件程序，包含了上图左边的3 个层次（标为灰色部分），所以实际的单片机程序只需要处理FAT 文件系统层，并且即使这一层也可以由CH375 的U 盘文件级子程序库实现。

如果不需要处理文件系统，也就是不处理上图左边的最顶层，那么CH375 直接提供了数据块的读写接口，以512字节的物理扇区为基本读写单位，从而将USB存储设备简化为一种外部数据存储器，单片机可以自由读写USB 存储设备中的数据，也可以自由定义其数据结构。

由于计算机将USB 存储设备组织为文件系统，为了方便单片机通过USB移动存储设备与计算机之间交换数据，单片机也可以将USB 存储设备组织为文件系统，也就是处理上图左边的最顶层。

CH375以C语言子程序库提供了USB存储设备的文件级接口，这些应用层接口API包含了常用的文件级操作，可以移植并嵌入到各种常用的单片机程序中。

CH375 的U 盘文件级子程序库具有以下特性：支持常用的FAT12、FAT16 和FAT32 文件系统，磁盘容量可达100GB 以上，支持多级子目录，支持8.3 格式的大写字母和中文文件名，可以支持小写字母或者长文件名，支持文件打开、新建、删除、读写以及搜索等。

CH375 的文件级子程序库需要大约600 字节的随机存储器RAM 作为缓冲区。以普通的MCS-51 单片机为例，文件系统的全部子程序有4KB 到8KB 代码，并且需要大约80 字节的内部RAM 和512 字节的外部RAM 作为缓冲区。有关U 盘文件级子程序库的详细信息请参考CH375

评估板的说明。

文件级子程序库的所有API 在调用后都有操作状态返回，但不一定有应答数据。有关API 参数的说明请参考CH375HF?.H，主要子程序如下：

初始化CH375 芯片：CH375Init

查询U 盘是否准备好：CH375DiskReady

查询U 盘容量：CH375DiskSize

查询U 盘信息（总容量及剩余容量）：CH375DiskQuery

打开文件：CH375FileOpen

枚举或者搜索文件：CH375FileEnumer

关闭文件：CH375FileClose

新建文件：CH375FileCreate

删除文件：CH375FileErase

以扇区为单位从文件读数据：CH375FileReadX

以扇区为单位向文件写数据：CH375FileWriteX

以扇区为单位移动文件指针：CH375FileLocate

查询文件属性（属性/日期/时间/长度）：CH375FileQuery

设置文件属性（属性/日期/时间/长度）：CH375FileModify

以字节为单位从文件读数据：CH375ByteRead

以字节为单位向文件写数据：CH375ByteWrite

以字节为单位移动文件指针：CH375ByteLocate

## 5.4 U 盘读写模块的并口说明

### 5.4.1 接口定义

模块具有两个外部接口：P1 是USB 插座，可以直接插入U 盘或者通过USB 延长线连接U 盘，当进行程序升级或者重新配置时应该通过USB 对连线连接计算机的USB端口；P2 是16脚的双排针或者插座，用于连接单片机系统。

标准版模块可以由工具软件设置，使之工作于并口方式或者串口方式。

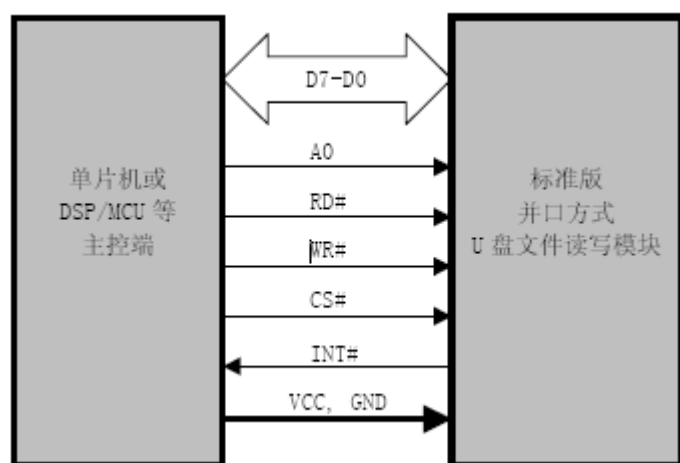
有关串口的引脚和协议说明，请参考CH375HMS.PDF 文档。

在并口方式下，外部系统的单片机与模块的P2 端口相连接。模块可以通过8 位数据线D0~D7、读RD#、写WR#、片选CS#信号线直接挂在单片机的8位系统总线上，除此之外，模块的中断输出INT#还应该连接到单片机的一个I/O 引脚（查询方式）或者中断输入引脚（中断方式）。

模块的被动并口由双向缓冲接口芯片CH421A 提供，单片机与模块之间的数据交换在CH421A 芯片的68 字节缓冲区中进行。当单片机系统向模块读写数据时，首先应该向模块的索引端口写入索引地址0~67，然后读写数据端口，每次读写一个字节数据后，CH421A自动将索引地址加1，以便单片机直接读写下一地址的数据。详细的操作步骤及时序可以参考CH421A 芯片手册中关于X 端口的说明。

引脚号	引脚名称	类型	并口方式下的 P2 端口引脚说明
16	GND	电源	公共接地端
15	VCC	电源	5V 正电源输入端，电源供给电流大于 100mA，需要外接电源退耦电容，容量不小于 200uF
14	CS#	输入	片选控制输入，低电平有效
13	NC	保留	保留引脚，禁止连接，禁止电平变化 (注：在旧版模块中是启动信号输入，下降沿有效)
12	WR#	输入	写选通输入，低电平有效
11	RD#	输入	读选通输入，低电平有效
10	A0	输入	地址信号线，为 0 时指向索引口，为 1 时指向数据口
9	INT#	输出	中断请求输出，低电平有效，带上拉开漏输出
1~8	D0~D7	双向三态	8 位双向数据总线，内置弱上拉电阻

注：并口连线较长时，需要特别考虑CS#和RD#以及WR#等控制线的串扰问题，解决方法是对关键信号进行隔离和屏蔽、增加关键信号的驱动能力、减小公共地线的连接电阻等。



### 5.4.2 参考电路

以下参考电路都是以MCS51 单片机为例，其它类型的单片机可以参考修改。

#### ① 并口连接，INT#中断通知

示例程序的应用环境就是采用这种连接方式，有关与模块连接的端口请参考连接电路图

②中的P2端口。模块的INT#连接到单片机的中断引脚INT0（或者INT1），如果使用查询方式，那么可以用普通I/O 引脚代替INT0 引脚。模块的A0 可接单片机的A0 或A8。

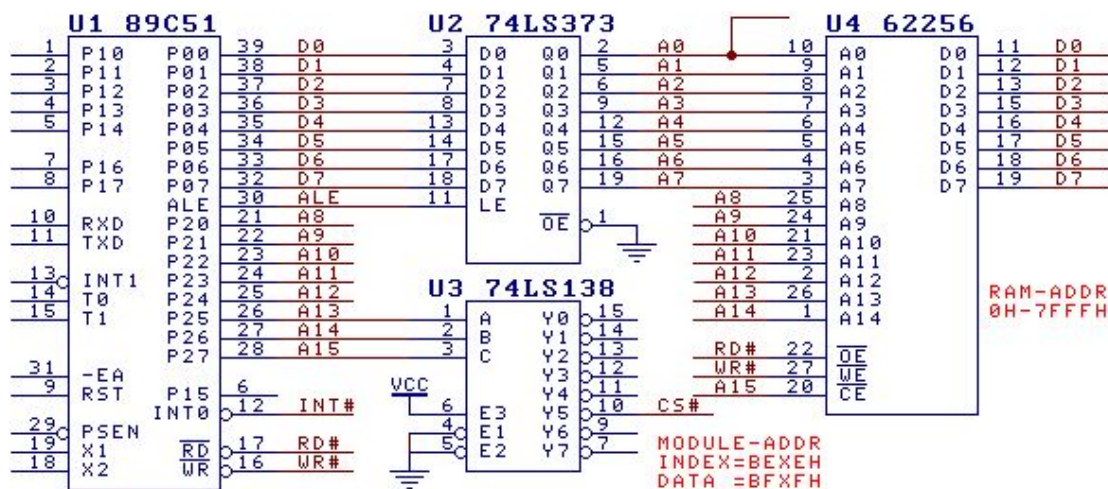


图 12

#### ② 并口连接，INT#查询状态

图中的P2 端口就是连接模块的端口。由于模块的CS#连接到单片机的A15引脚，所以其端口地址范围为0000H-7FFFH，如果需要连接外部RAM，那么其地址应该在8000H 以上才能避免冲突。如果不连接外部RAM，例如采用具有1KB 以上内部RAM的单片机，那么应该避免在读取内部RAM时与模块地址冲突。当然，最佳方法是参考连接电路图①由地址译码电路U3 产生模块的CS#信号。

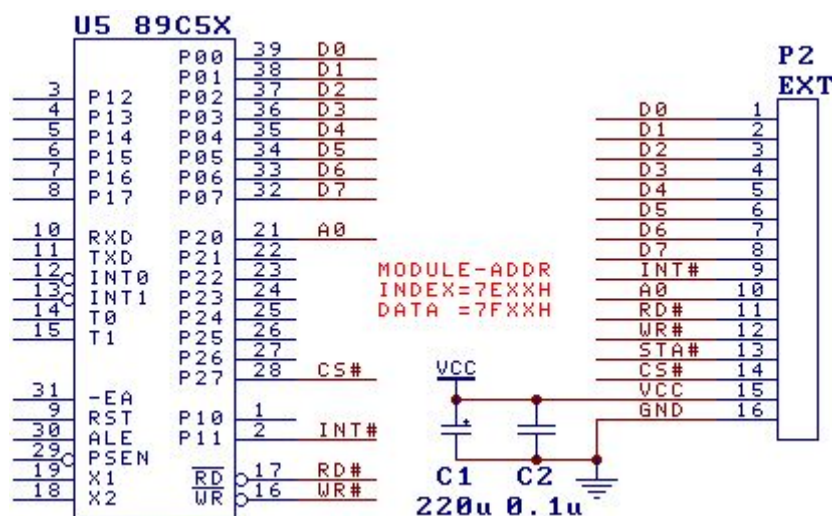


图 13

### 5.4.3 接口操作

基本操作步骤是，单片机系统将命令码、后续参数长度（因为各命令码所需要的参数不等长）和参数写给模块，并通知其启动操作，模块执行完成后以中断方式通知单片机，并返回操作状态和操作结果。

因为接口操作看起来比较复杂，所以实际过程可以参考随模块一起提供的几个示例程序，直接用其中的ExecCommand 子程序就可以了，不必理解下面的接口步骤说明。

该操作步骤适用于V3.1 及以上版本的模块，与之前版本的模块的操作步骤不同。

并口方式的操作步骤是（请参考示例程序中的ExecCommand 子程序）：

① 基本概念：并口数据交换通过模块的CH421A 芯片进行，外部单片机与模块交换数据实际上就是单片机存取CH421A 芯片的内部缓冲区。当模块挂在外部单片机的系统总线上时，模块需要占用外部单片机系统的两个I/O 地址，模块的I/O地址就是CH421A 芯片的I/O地址，低地址为索引端口，高地址为数据端口。CH421A 芯片提供66 字节的双口SRAM 缓冲区（地址为00H-41H）以及两个字节的单向缓冲区（地址为42H 和43H）。模块的通讯协议约定，从地址PARA\_BUFFER\_ADDR 即00H 开始的64 个字节是数据块缓冲区，地址范围是00H-3FH；地址PARA\_COMMAND\_ADDR 即40H 是操作命令码的单元地址，该单元由单片机在操作前设置，由模块在操作完成后清除；地址PARA\_STATUS\_ADDR即41H 是操作结果状态码的单元地址，该单元由模块在操作完成后设置；地址PARA\_CMD\_LEN\_ADDR 即42H 是命令包长度的单元地址，该单元低7位为后续参数的长度，最高位用于通知模块，为1 则激活/继续，为0 则结束/应答，该单元由单片机在操作前设置；地址PARA\_STS\_LEN\_ADDR 即43H是状态包长度的单元地址，该单元低7 位为后续参数的长度，由模块在操作完成后设置以返回给单片机。

② 单片机系统将命令码写到CH421A 的索引地址PARA\_COMMAND\_ADDR中，将命令的后续参数写到CH421A 的索引地址PARA\_BUFFER\_ADDR 开始的缓冲区中，并将这些后续参数的长度写到CH421A 的索引地址PARA\_CMD\_LEN\_ADDR 中并设置该单元最高位为1 以激活操作，有些命令不需要任何参数，那么参数的长度就应该是0（由于最高位为1，实际数据是80H）。

③ 默认情况下，PARA\_CMD\_LEN\_ADDR 的最高位为0，当单片机系统设PARA\_CMD\_LEN\_ADDR 的最高位为1 后，该0-1 转变将通知模块启动操作。

④ 模块检测到PARA\_CMD\_LEN\_ADDR 的0-1转变后，从CH421A读取命令码及可选的参数，



然后分析命令码并执行。

⑤ 对于以字节为单位的文件数据块读写操作请跳过此步骤，对于以扇区为单位的文件数据块读写操作还应该有以下步骤，每读写一个扇区的数据，以下过程就会重复8次，每次传输64字节的数据，共8次可以传输一个扇区的数据。每次传输过程必须在2mS时间之内完成，否则有些U盘可能会工作不正常。

如果是CMD\_FileRead 或CMD\_FileReadLast 命令，模块将请求读取数据的状态码USB\_INT\_DISK\_READ 写到CH421A的索引地址PARA\_STATUS\_ADDR，然后模块的INT#引脚输出低电平。单片机系统检测到INT#的低电平后，应该从CH421A的PARA\_STATUS\_ADDR读取状态码，接着设置PARA\_CMD\_LEN\_ADDR的最高位为0（作为中断应答），然后从CH421A的索引地址PARA\_BUFFER\_ADDR开始，连续取走64字节的数据块，取走64字节数据后，单片机系统应该设置PARA\_CMD\_LEN\_ADDR的最高位为1，通知模块已经取走数据，可以继续。

如果是CMD\_FileWrite 命令，模块将请求写入数据的状态码USB\_INT\_DISK\_WRITE 写到CH421A的索引地址PARA\_STATUS\_ADDR，然后模块的INT#引脚输出低电平。单片机系统检测到INT#的低电平后，应该从CH421A的索引地址PARA\_STATUS\_ADDR读取状态码，接着设置PARA\_CMD\_LEN\_ADDR的最高位为0（作为中断应答），然后从CH421A的索引地址PARA\_BUFFER\_ADDR开始，连续写入64字节的数据块，写完数据后，单片机系统应该设置PARA\_CMD\_LEN\_ADDR的最高位为1，通知模块已经写入数据，可以继续。

如果模块在数据读写过程中检测到错误，那么模块将读写数据块失败重试状态码USB\_INT\_DISK\_RETRY写到CH421A的索引地址PARA\_STATUS\_ADDR，然后模块的INT#引脚输出低电平。单片机系统检测到INT#的低电平后，应该从CH421A的索引地址PARA\_STATUS\_ADDR读取状态码，接着设置PARA\_CMD\_LEN\_ADDR的最高位为0（作为中断应答），然后从CH421A的索引地址PARA\_BUFFER\_ADDR开始连续读取两个字节的的数据，之后，单片机系统应该设置PARA\_CMD\_LEN\_ADDR的最高位为1，通知模块可以继续。读取的两个字节是一个16位的数据，指定需要回改指针的字节数，大端时高字节在前，小端时低字节在前。用户端程序接收到USB\_INT\_DISK\_RETRY 状态码后，应该根据该16位数据回改文件数据缓冲区指针，以便重新发送或接收数据。

⑥ 模块执行完成，将操作状态码写到CH421A的索引地址PARA\_STATUS\_ADDR。如果状态码为操作成功ERR\_SUCCESS，那么模块还将后续的结果数据的总长度写到CH421A的索引地址



PARA\_STS\_LEN\_ADDR中，并从CH421A 的索引地址PARA\_BUFFER\_ADDR 开始写入结果数据。有些命令执行后没有结果数据返回，那么结果数据的长度就会是0。

⑦ 模块将PARA\_COMMAND\_ADDR 清零，并使INT#引脚输出低电平，通知单片机系统该命令操作完成。

⑧ 单片机系统收到INT#的低电平中断（或者下降沿中断，或者定期查询发现INT#为低电平）后，从CH421A 读取返回的状态码及可选的结果数据，然后设置PARA\_CMD\_LEN\_ADDR的最高位为0（作为中断应答）。当模块收到下一个启动操作信号（指PARA\_CMD\_LEN\_ADDR 最高位的0-1 转变）或者PARA\_CMD\_LEN\_ADDR 的最高位恢复为0 时，就会将模块的INT#引脚恢复为高电平，撤消对单片机的中断请求。至此，一个操作完成。

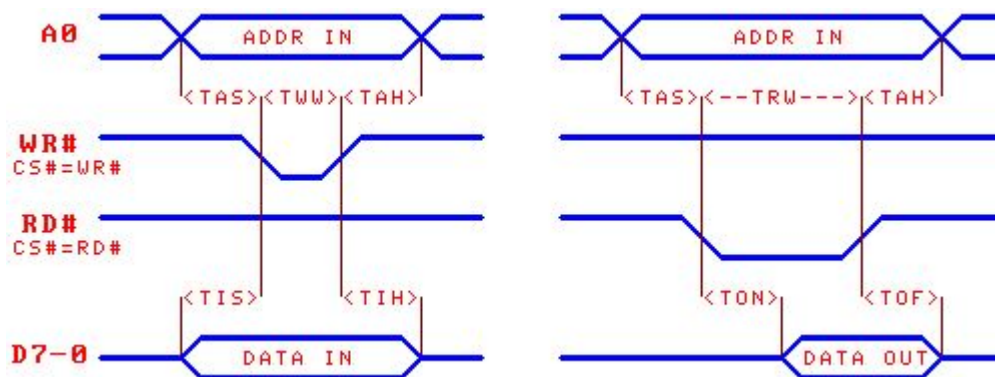
#### 5.4.4 时序接口

测试条件：TA=25℃，VCC=5V

（RD 是指RD#信号有效并且CS#信号有效，RD#=CS#=0 执行读操作）

（WR 是指WR#信号有效并且CS#信号有效，WR#=CS#=0 执行写操作）

名称	参数说明	最小值	典型值	最大值	单位
TWV, TRW	读/写选通脉冲 RD/WR 的宽度	80			nS
TPC	并口读操作或者写操作的周期	100			nS
TAS	RD/WR 前的地址建立时间	10			nS
TAH	RD/WR 后的地址保持时间	6			nS
TIS	RD/WR 前的数据建立时间	3			nS
TIH	RD/WR 后的数据保持时间	5			nS
TON	读选通 RD 有效到数据输出有效	0	50	70	nS
TOF	读选通 RD 无效到数据输出无效	0		25	nS
TINT	收到撤消条件到 INT#恢复高电平			5	uS



## 5.5 U 盘读写的串口说明

### 5.5.1 接口定义

模块具有两个外部接口：P1 是USB 插座，可以直接插入U 盘或者通过USB 延长线连接U 盘，当进行程序升级或者重新配置时应该通过USB 对连线连接计算机的USB 端口；P2 是16 脚的双排针或者插座，用于连接单片机系统。

在串口方式下，单片机与模块的P2 端口相连接，只需要使用高8 脚（第9 脚到第16脚），其余引脚可以不连接。模块可以通过串行输入SIN、串行输出SOUT 连接到单片机的异步串口，除此之外，启动输入STA#还应该连接到单片机的一个I/O 引脚，而中断输出INT#可以根据需要决定是否连接到单片机的中断引脚。

模块的串口是1 位起始位、8 位数据位、1位停止位的异步串口，串口的通讯波特率可以在功能配置时设定，如果未设定那么默认是4800bps（与晶体X2 的频率有关），单片机系统可以根据需要通过CMD\_BaudRate 命令设定更高的波特率。如果在功能配置时选择检查串口超时，那么在通过串口输入命令包时，模块会检查串口数据输入超时，如果连续两个数据字节之间的间隔大于串口输入超时时间，则模块将放弃该命令包。

4+1 线串口是指GND、SIN、SOUT、STA#和可选的INT#，P2 引脚定义如下：

引脚号	引脚名称	类型	串口方式下的 P2 端口引脚说明
16	GND	电源	公共接地端
15	VCC	电源	5V 正电源输入端，电源供给电流大于 100mA，需要外接电源退耦电容，容量不小于 200uF
14	保留		悬空或者接高电平，建议接高电平
13	STA#	输入	启动信号输入，下降沿有效，内置上拉电阻
12	SIN	输入	串行数据输入，内置上拉电阻
11	SOUT	输出	串行数据输出，带上拉开漏输出
10	保留		悬空
9	INT#	输出	中断请求输出，低电平有效，带上拉开漏输出
1~8	保留		必须悬空

为了节约单片机的I/O 引脚，模块还支持三线制串口，在这种方式下，单片机与模块之间只需要连接SIN 和SOUT 两根信号线及公共地线，单片机通过串口发送两个同步码字节（57H、ABH）代替原来向模块的STA#引脚提供的启动信号，实现与模块的命令同步。在通过串口输入两个同步码字节时，模块会检查串口数据输入超时，如果连续两个数据字节之间的间隔大于串口输入超时时间，则模块将放弃该同步码及命令包。

3 线串口是指GND、SIN、SOUT，P2 引脚定义如下：

引脚号	引脚名称	类型	串口方式下的 P2 端口引脚说明
16	GND	电源	公共接地端
15	VCC	电源	5V 正电源输入端，电源供给电流大于 100mA，需要外接电源退耦电容，容量不小于 200uF
14	保留		悬空或者接高电平，建议接高电平
13	保留		悬空或者接高电平
12	SIN	输入	串行数据输入，内置上拉电阻

11	SOUT	输出	串行数据输出，带上拉开漏输出
10	保留		悬空
9	INT#	输出	悬空
1~8	保留		必须悬空

### 5.5.2 参考电路

以下参考电路都是以MCS51 单片机为例，其它类型的单片机可以参考修改。

#### ① 串口连接，串口等待状态

图中的P3 端口就是连接模块的端口。由于是串口连接，所以连接信号线较少，只需要三个信号线：SIN，SOUT 和STA#，模块的中断信号线INT#是可选的，如果采用中断通知方式，那么可以连接INT0 或者INT1，否则不必连接模块的INT#信号线。建议参考该图，将模块端口的14 脚（原并口的CS#引脚）接VCC，避免并口被选中。

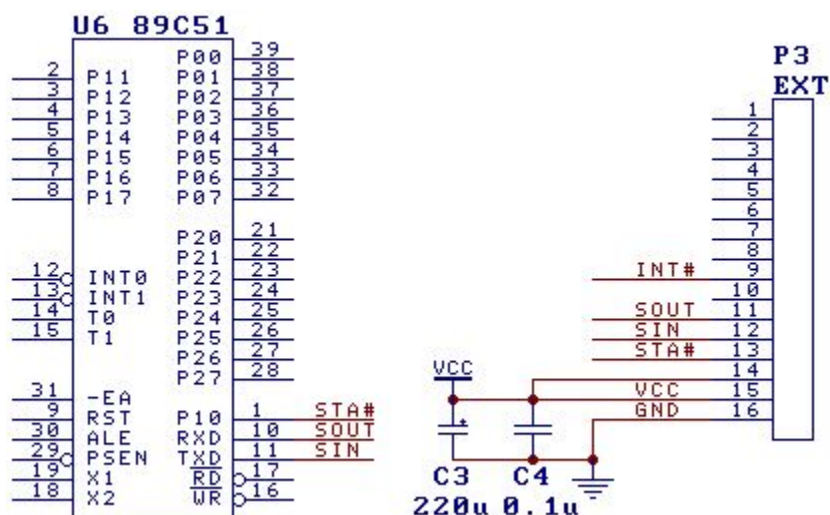


图 14

#### ② 三线制串口连接，串口等待状态

图中的P4 端口就是连接模块的端口。由于是三线制串口连接，所以连接信号线最少，只



基本操作步骤是，单片机系统将命令码、后续参数长度（因为各命令码所需要的参数不等长）和参数写给模块，并通知其启动操作，模块执行完成后以中断方式通知单片机，并返回操作状态和操作结果。

串口方式的操作步骤是（请参考示例程序中的ExecCommand 子程序）：

- 28 -

不需要任何参数，那么参数的长度就应该是0。

- ④ 模块检测到STA#引脚的下降沿后，从串口依次接收命令码及可选的参数，然后分析命令码并执行。如果是三线制串口，那么模块以在20mS 之内收到同个同步码字节作为STA#下降沿的代替信号。
- ⑤ 对于以字节为单位的文件数据块读写操作请跳过此步骤，对于以扇区为单位的文件数据块读写操作还应该有以下步骤，每读写一个扇区的数据，以下过程就会重复8 次，每次传输64 字节的数据，共8 次可以传输一个扇区的数据。如果是CMD\_FileRead 命令，模块将INT#引脚输出低电平，然后通过串口输出请求读取数据的状态码USB\_INT\_DISK\_READ，状态码输出完成后，模块将INT#引脚恢复为高电平，然后从串口依次输出64 字节的数据。单片机系统应该在收到状态码后，再从串口连续接收64 字节的数据块。如果是CMD\_FileWrite 命令，模块将INT#引脚输出低电平，然后通过串口输出请求写入数据的状态码USB\_INT\_DISK\_WRITE，状态码输出完成后，模块将INT#引脚恢复为高电平，然后等待从串口输入数据。单片机系统应该在收到状态码后，从串口连续输出64 字节的数据块。如果模块在数据读写过程中检测到错误，那么模块将INT#引脚输出低电平，然后通过串口输出读写数据块失败重试状态码USB\_INT\_DISK\_RETRY，状态码输出完成后，模块将INT#引脚恢复为高电平，然后从串口依次输出两字节的数据。单片机系统应该在收到状态码后，再从串口连续接收两字节的数据。这两字节是一个16 位的数据，指定需要回改指针的字节数，大端时高字节在前，小端时低字节在前。用户端程序接收到USB\_INT\_DISK\_RETRY 状态码后，应该根据该16 位数据回改文件数据缓冲区指针，以便重新发送或接收数据。
- ⑥ 模块执行完成，将INT#引脚输出低电平，通知单片机系统命令操作完成，“状态码开始发送”，然后通过串口输出操作状态码，状态码输出完成后，模块将INT#引脚恢复为高电平。如果状态码为操作成功ERR\_SUCCESS，那么模块还从串口依次输出后续的结果数据的长度、以及可选的结果数据。有些命令执行后没有结果数据返回，那么结果数据的长度就会是0。
- ⑦ 单片机系统收到INT#的下降沿中断（或者定期查询发现INT#为低电平，或者未连接模块的INT#引脚，而是从串口收到状态码）后，从串口获得返回的状态码及可选的结果数据。如果是三线制串口，那么由于未连接模块的INT#引脚，所以单片机只能以串口收到操作状态码及结果作为操作结束。

⑧ 单片机系统根据需要可以转到步骤②发出下一个操作命令。

#### 5.5.4 接口时序

测试条件：TA=25℃，VCC=5V

名称	参数说明	最小值	典型值	最大值	单位
TSTA	STA#输入启动信号脉冲宽度	1	2		μS
TT0	串口输入数据的间隔超时时间	25	40	100	mS

## 6 软件设计

为了使上位 PC 机能够直接读取该读写器写入 U 盘的数据，数据存储按照 FAT32 文件管理方式存储，其中涉及 USB 和 UFI 协议以及 FAT32 文件存储格式。

### 6.1 USB 协议

USB（通用串行总线）用于将 USB 接口的外围设备（device）连接到主机（host），实现二者之间数据传输的外部总线结构，是一种快速、灵活的总线接口，USB 的传输类型有控制（control）、批量（bulk）、中断（interrupt）和同步（synchronous）传输 4 种，它最大的特点是易于使用，即插即用，主要是用在中速和低速的外设。

控制数据用于在 USB 接入总线时对其进行配置，其他的驱动软件可以根据具体的应用来选择使用控制传输，这种数据传输不会丢失数据。

典型的批量数据包括象使用打印机或扫描仪时所出现的大数据量的数据，这种批量数据是连续的，通过在硬件中实现差错检测功能，并且有选择地进行一定的应进重试操作，可以在硬件层次上保证数据的可靠交换。

由设备自发产生的数据传输是中断数据传输，这类数据传输可以由 USB 设备在任意时刻发起，而且 USB 总线以不低于设备说明的速率进行传输。

同步数据在产生、传送和处理过程中是连续的和实时的，在稳定的同步数据发送和接收速率中包含了相应的时钟信息，为了保持定时关系，同步数据必须按照接收的速率进行传输。

### 6.2 BULK\_ONLY 和 UFI 协议

USB 设备分为 5 大类，即显示器、通信设备、音频设备、人机输入和海量存储。通常所用的 U 盘、移动硬盘均属于海量存储类。海量存储类的规范中包括 4 个独立的子规范，即 CBI 传输、Bulk-Only 传输、ATA 命令块、UFI 命令规范。前两个协议定义了数据/命令/状态在 USB 总线上的传输方法，Bulk-Only 传输协议仅仅使用 Bulk 端点传送数据/命令/状态，CBI 传输协议则使用 Control/bulk/interrupt 三种类型的端点进行数据/命令/状态的传送。后两个协议定义了存储介质的操作命令，ATA 协议用于硬盘，UFI 协议则针对 USB 移动存储，U 盘读

写器的设计遵循 Bulk-Only 传输协议和 UFI 命令规范。UFI 命令块规范是针对 USB 移动存储而制定的，它总共定义了 19 个 12 字节长度的操作命令。

Bulk-Only 事务以主机向设备发送 CBW（Command Block Warp）包，并以建立相应的数据传输开始的，设备接收到 CBW 包，检查并解释它，试图满足主机的要求，并通过 CSW（Command State Wrap）包向主机返回状态信息。

CBW 是主机通过 Bulk-Out 端点向设备发送的命令块包，在 CBW 中使用方向位和数据传输长度域指明期待的传输，CBW 必须起始于包边界，并且必须以 31 字节的短包传输结束，相继的数据包和 CSW 包必须开始于一个新的包边界，所有的 CBW 包必须按低字节在前的次序传输。

CBW 包结构如图 16 所示，各域含义如下：

字节 \ 位	7	6	5	4	3	2	1	0
0~3	命令块包标识							
4~7	命令块标记							
8~11	数据传输长度							
12	命令块标旗							
13	预留				逻辑单元号			
14	预留				CBWCB长度			
15~30	CBWCB							

图 16 CBW 包结构如图

- 1) 命令块包标识。CBW 包标记，表明这是一个 CBW 包，这个域的值 43425355H。
- 2) 命令块标记。当设备返回相应的 CSW 包时，必须使命令状态标记域的值与此值相同。
- 3) 数据传输长度。指明命令执行期间在 Bulk 端点上传数据的字节长度，如果这个域的值是 0，则在 CBW 和 CSW 之间设备和主机不传输任何数据，并且设备将忽略在命令块标旗域中的方向位的值。
- 4) 命令块标旗。方向位规定了 Bulk 端点数据传输的方向，其他位预留。
- 5) 逻辑单元号。指定命令块被发送到的逻辑单元号，如果设备不支持多个逻辑单元号，则主机将这个域设置为 0。
- 6) CBWCB 长度，定义了 CBWCB 的有效长度，合法值为 1-16。



7) CBWCB。由设备执行的命令，由设备解释。

CSW 向主机表明来自于 CBW 包的命令块的执行状态。设备收到 CBW 包解析处理后将通过 Bulk-In 端点发送一个 CSW 包。

CSW 开始于包边界，并以 13 字节的短包结束，结构如图 17 所示，各域含义如下：

字节 \ 位	7	6	5	4	3	2	1	0
0~3	命令状态包标识							
4~7	命令状态标记							
8~11	数据残余							
12	命令执行状态							

图 17 命令状态包结构

- 1) 命令状态包标识。CSW 包的标记，表明这是一个 CSW 包，这个域的值 53425355H。
- 2) 命令状态标记。次域的值域 CBW 包的命令块标记相同。
- 3) 数据残余。实际数据传输量与 CBW 包中规定的数据传输长度的差值。
- 4) 命令执行状态，表明命令成功或失败信息，如果命令执行成功，则设备将设置此域的值 0，非 0 值；则表明失败或错误。

UFI 是针对 USB 移动存储而制定的命令块协议，它规定了主机和设备进行信息交换所使用的命令块、数据和状态信息，Bulk-Only 传输协议定义了传输这些信息的方法，其中 UFI 命令块是封装在 CBW 包中的 CBWCB，设备通过读取 CBWCB 确定具体要执行何种操作命令（如读命令），如何完成这个命令（如从闪存的哪个地址读，需要读取的长度），设备将命令的执行状态封装成 CSW 返回给主机。

UFI 用于大多数命令的 12 字节命令块的描述，结构如图 4 所示，其中各参数意义如下：

- 1) 操作命令代码。指明所需要执行的操作命令；
- 2) 逻辑单元号。指明命令将发送到哪个逻辑单元，如果设备只有一个逻辑单元，则此域的值 0。
- 3) 逻辑块地址。命令操作的起始地址。

- 4) 传输长度，指明请求传输的数据量，通常以“扇区”作单位，但是有几个命令是以“字节”作单位的，对于这些命令，传输长度域可以以不同的名字标识，若此域的值 0，则表面没有数据需要传输。
- 5) 参数列表长度，用于指定发送到设备的字节数，这个域典型的应用于发送到设备的参数命令块（如模式参数、诊断参数等），若此域的值 0，则表面没有数据需要传输。
- 6) 分配长度，指明主机已经分配的用于返回数据的最大字节长度，若此值为 0，则表明没有数据需要传输。

## 6.3 FAT32

FAT 是 Microsoft 较早推出的文件系统，具有高度的兼容性，目前仍然广泛应用于个人电脑尤其是移动存储设备中，FAT 由引导扇区、FAT1 表、FAT2 表、目录和文件区组成（其中 FAT2 表是 FAT1 表的备份）。磁盘的管理是以扇区为单位的，而移动存储设备则是以块为单位的，FAT 将块映射成扇区，原理相同，FAT 将磁盘空间以一定数目的扇区为单位进行划分，这样的单位成为“簇”。通常情况下，每扇区 512 字节的原则是不变的，簇的大小一般是  $2n$ （ $n$  为整数）个扇区的大小。所以以簇为单位而不以扇区为单位进行磁盘的分配，是因为当区分容量较大时，采用 512 字节的扇区管理，会增加 FAT 表的项数和大文件存取消耗，使文件系统效率不高。

引导扇区 DBR (DOS Boot Record) 通常占用分区的第 0 扇区，共 512 字节，FAT 表紧随其后。DBR 的第一部分是一个 x86 跳转指令、厂商标志和操作系统版本号，接下来的从偏移 0x0B 开始的是一段描述能够使可执行引导代码找到相关参数的消息，通常称之为 BPB。最后是引导程序代码以及扇区结束标志。BPB 中记录了扇区大小、簇的扇区数、保留扇区数、FAT 表大小和文件系统类型等重要参数，用于文件的索引和定位计算。

## 6.4 软件系统模块

USB 读写器软件由主模块、USB 模块、BULK 模块、FAT 模块和中断处理模块组成。各模块之间相互协调调用，共同完成对 U 盘文件的读写创建。

USB 模块负责检测 USB 设备的移入/移出，对插入的设备进行枚举，分配设备地址端点号，配置设备接口端点描述符，建立 BULK\_ONLY 输入/输出通道，BULK 模块则在已建立的 BULK\_ONLY 输入/输出通道发送 CBW 数据包，并接收 CSW 数据包，通过 CBW 中嵌入的 CBWCB 信息确定对 U 盘读/写操作的扇区位置和大小，FAT 模块主要完成簇和扇区间的索引定位，即在 U 盘内寻址，利用 BPB 中的参数计算给定扇区的所在簇以及进入休眠状态，等待外部中断唤醒。

中断处理模块是在中断唤醒后根据中断类型进入不同功能子模块的，也是整个软件系统的核心部分。中断模块接收到串口或并口数据后，根据内部命令字要求进行解析，命令字要求如图 5 所示，串口或并口数据接收流程如图所示。

字节 \ 位	7	6	5	4	3	2	1	0
0	操作命令代码							
1	逻辑单元号				预留			
2~5	逻辑块地址							
6	预留							
7~8	传输长度或参数列表长度或分配长度							
9~11	预留							

图 18 UFI 结构

类型 \ 位	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
命令字	1	1	命令值						命令参数字节数或数据长度							
数据字	1	0	预留		12位数据				12位数据							

图 19 命令与数据字结构

## 7 总结

此次毕业设计是我们从大学毕业生走向未来工程师重要的一步。从最初的选题，开题到计算、绘图直到完成设计。其间，查找资料，老师指导，与同学交流，反复修改图纸，每一个过程都是对自己能力的一次检验和充实。

通过这次实践，我了解了单片机读写 U 盘的用途及工作原理，熟悉了单片机读写 U 盘的设计步骤，锻炼了工程设计实践能力，培养了自己独立设计能力。此次毕业设计是对我专业知识和专业基础知识一次实际检验和巩固，同时也是走向工作岗位前的一次热身。

毕业设计收获很多，比如学会了查找相关资料相关标准，分析数据，提高了自己的绘图能力，懂得了许多经验公式的获得是前人不懈努力的结果。同时，仍有很多课题需要后辈去努力去完善。

但是毕业设计也暴露出自己专业基础的很多不足之处。比如缺乏综合应用专业知识的能力，对材料的不了解，等等。这次实践是对自己大学四年所学的一次大检阅，使我明白自己知识还很浅薄，虽然马上要毕业了，但是自己的求学之路还很长，以后更应该在工作中学习，努力使自己 成为一个对社会有所贡献的人。

## 致谢

深深感谢我的指导老师潘文诚老师，在毕业设计期间对我的悉心指导和亲切关怀。同样要感谢各科的老师在四年来的学习生活中给予我的精心指导、照顾和鼓励。衷心感谢所有那些给过我帮助的老师、同学和朋友。

邵旭峰

2008-6-10 于杭州

## 文献参考

- [1] USB 总线接口芯片 CH375 中文手册[Z] 南京沁恒电子有限公司 2004
- [2] USB-HOST 芯片 CH375 的 U 盘文件级子程序库说明[Z] 南京沁恒电子有限公司 2004
- [3] 肖踞雄 翁铁成 宋中宋 USB 技术及应用设计[M] 北京 清华大学出版社 2003
- [4] 邓清涛 杨鹏 基于 CH375 的单片机 U 盘控制器的设计与实现 重庆大学自动化学院 2007
- [5] 刘井权 王宪 基于单片机的 USB 接口的设计[J] 自动化仪表 2006
- [6] 陈启美 丁传锁 计算机 USB 接口技术[M] 南京 南京大学出版社 2003
- [7] 马伟 计算机 USB 系统原理及其主/从机设计[M] 北京 北京航空航天大学出版社 2004
- [8] 孙霞 基于单片机的大容量数据存储技术的应用研究 微计算机信息 2006
- [9] 张峰峰 嵌入式 U 盘读写器接口技术和系统设计 桂林电子科技大学 2007
- [10] 杨全玖 基于 CH375 多通道 USB 接口与 PC 机通信的实现 微计算机信息 2007
- [11] Don Anderson. Universal Serial Bus System Architecture [M].Addison Wesley Longman, Inc, 2000
- [12] Phoenix Technologies Ltd Universal Serial Bus 1996

