

## 基于通用串行总线 and 有限状态机的数据发送系统设计

杨晓光<sup>1,2</sup>

(1 中国科学院 声学研究所, 北京 100190; 2 中国科学院 研究生院, 北京 100190)

**摘 要:** 设计了基于高速数据接口和有限状态机逻辑控制的数据发送系统. 可实现通过 PC 机发送测试数据和保存的实验数据至数据处理系统的应用, 便于数据处理系统特别是其中硬件部分的调试和检测. 灵活性好, 通过修改 FIFO 接口数据的读写方式可以适应多种不同系统数据接口.

**关键词:** 有限状态机; 系统测试; 通用串行总线

中图分类号: TN432

文献标识码: A

文章编号: 1000-7180(2010)04-0162-03

## Data Transmit System Base on USB and FSM

YANG Xiao-guang<sup>1,2</sup>

(1 Institute of Acoustics, Chinese Academy of Sciences, Beijing 100190, China;

2 Graduate School, Chinese Academy of Sciences, Beijing 100190, China)

**Abstract:** This paper designs a data transmit system base on USB and FSM logic control. It can be used for data process system especially hardware part debugging and testing conveniently by transmitting test data from PC. And with highly flexible, it can work with many hardware data interface by a few adjust on FIFO read and write schedule.

**Key words:** FSM; system test; USB

### 1 引言

在设计、调试和检测数字信号处理系统的过程中, 需要对数据传输存储的可靠性、数据处理的性能以模拟实验的方式进行评估. 单纯软件仿真的方法无法有效考察硬件系统的性能, 而外场实验成本高昂. 有效利用外场实验数据对处理系统进行模拟实验成为一种必然选择. 因此, 需要一种能够取代传感器组件将预先保存的实验数据直接导入数据处理系统进行模拟实验的设备. 对于存储在 PC 机上的实验数据, 利用 USB 高速数据接口导出是最为便利的方式. 有限状态机描述的逻辑可以有效地分配数据, 将数据文件分解成符合处理系统要求的数据格式.

### 2 设计方法

本数据发送装置需要模拟的数据采集系统描述如下: 传感器组采集的模拟信号经 A/D 转换后存入

FIFO 中, 当存满一帧后, 向 DSP 发出中断, 等待 DSP 读取, 经过一段预设的时间后, 清空 FIFO, 开始存入下一组数据<sup>[1]</sup>. 系统框图如图 1 所示.

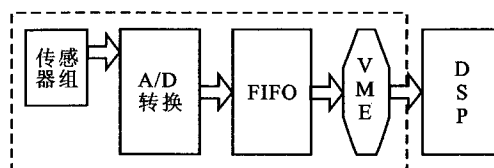


图 1 原数据采集系统框图

采集到的原始数据上传后以数据文件的形式保存在 PC 机中. 数据发送系统则要利用 USB 高速数据接口, 将存储于 PC 机中的实验数据分批存入与 DSP 相连的 FIFO 中, 并向 DSP 发中断, 等待 DSP 读取, 从而实现回放实验数据. DSP 的工作时序不做任何调整. 系统框图如图 2 所示.

#### 2.1 通用串行总线接口设计

通用串行总线传输速率高, 使用方便, 支持热插

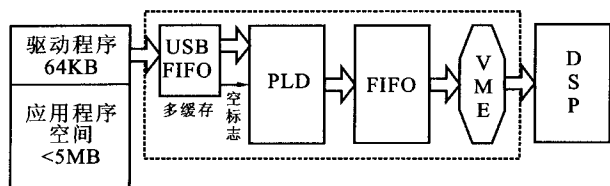


图2 数据发送系统框图

拔. 软件方面, 为 USB 设计的驱动程序和应用软件可自动启动, 无需干预. 且 USB 设备不涉及 IRQ 冲突问题, 单独使用自己的保留中断, 免去硬件配置的麻烦. 连接方式灵活, 连接线可达 5m, 如增加中继, 还可再延长, 实验操作非常便利.

本设计选用了 CYPRESS 公司的 EZ-USB FX2 作为 USB 接口的控制芯片. 其套件为用户提供一个固件函数库 (Ezusb.lib) 和固件框架 (Framework), 固件函数库提供了一系列的函数, 使用时只需在程序中包含 EZUSB.H 和 EZREGS.H 两个头文件, 并在项目中链接 Ezusb.lib 就可以直接使用固件库中的各个函数了. 在使用固件框架来进行特定的固件开发时, 固件框架已经实现了初始化、重枚举、源管理等功能. 只需要在固件架构的预留位置处添加代码, 就可以实现特定的功能. 本设计中只需要修改 periph.c 文件中的 TD\_Init() 函数. 将 FX2 芯片的端点 2 (EP2) 配置为  $512 \times 4$  的 4 级缓冲异步 FIFO, 以保证数据流的连续性<sup>[2-3]</sup>.

## 2.2 控制逻辑设计

由于原数据采集系统为上电启动后工作, 无启动控制信号, 只能在系统开始工作记录数据之后再发送数据文件. 因此, 无法在保存的全部数据中准确记录发送数据的起止. 为了在接收的数据中准确区分出发送的任务数据, 每个任务开始前系统设定为向 FIFO 中持续写入任务数据中不会出现的 FFFF 值作为给 DSP 的握手信号, 再通过手动控制 USB 驱动来启动数据文件的传送. 即当出现有效数据等待发送时发送数据, 任务开始前无数据待发时发送 FFFF. 由于 USB 系统的特点, PC 机上的 USB 驱动程序在内存中使用容量为 64KB 的空间. 而任务数据文件远大于 64K, 在发送同一个任务数据的过程中, 应用程序发送完一批 64KB 数据后需要重新填满 64KB 的内存空间, 这个填充过程需要一定时间, 且需要的时间并不为定值. 在这段时间内, 不应给 DSP 握手信号 FFFF; 只有当一个任务的数据文件发送完成后, 才需要给 DSP 握手信号 FFFF 以标志数据的结尾. 为了实现这种区别, 引入看门狗计时, 当系统发现等待时间超过 500ms 后仍没有数据填

充 64KB 的内存空间, 就认为本次任务已经完成, 并给出握手信号 FFFF 标志数据结束; 当系统等待不足 500ms 就有数据填充, 则认为是同一个任务中的数据. 这种设计, 保证了数据的连续性, 避免了引入伪噪声 FFFF 而将同一数据文件分拆<sup>[4-5]</sup>.

周期性产生的 clr\_n 信号用于控制写-读 FIFO 循环, 作为总体控制信号产生控制读取 USB 控制芯片 EP2 FIFO 和与 DSP 连接的 FIFO 的控制波形. 这两种波形由周期性产生的 clr\_n 信号同步. 系统流程如图 3 所示.

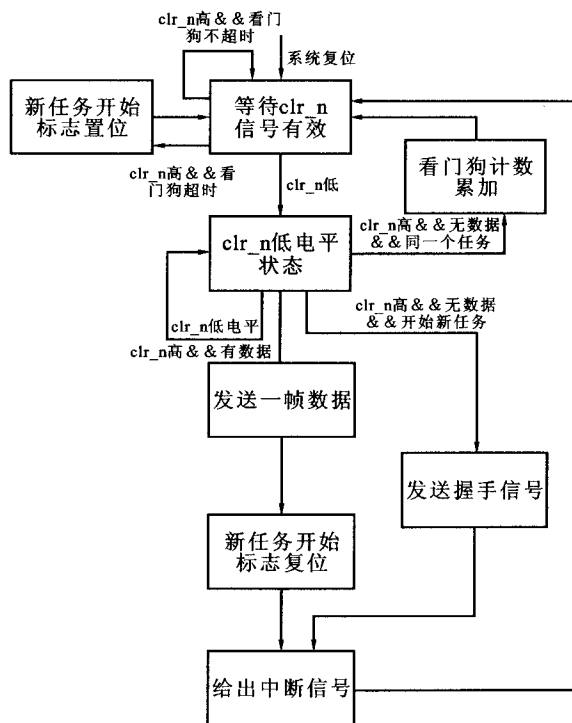


图3 数据发送系统控制流程图

## 2.3 状态机设计

实现逻辑控制的有限状态机需要实现以下 3 条基本功能.

- (1) 根据系统硬件设计, 系统时钟为 12MHz.
- (2) 采用计数分频的方法, 利用系统时钟产生周期为  $57\mu\text{s}$ , 占空比为 56/1 的 clr\_n 信号 (与原数据采集系统相匹配).
- (3) 产生各种控制信号 (包括控制 USB FIFO 和与 DSP 相连 FIFO 的信号).

需要分辨出当前是否有数据等待发送, 每个任务开始前没有数据时向 FIFO 填充 FFFF 值. 驱动程序的缓存为 64KB, 当前一个 64KB 发送完成后重新填满驱动程序 64KB 缓冲需要一定的时间  $t_1$ .

根据 USB FIFO 的空标志 EF 是否有效来判断

是否存在要发送的数据.造成 EF 有效(表示当前无待发数据)有两种可能:(1)由同一个任务内  $t_1$  的存在造成的;(2)一个任务发送完成后,等待手动控制应用软件启动下一组数据的发送任务,两次任务之间的空闲时间  $t_2$ .

通过分析可以看出,状态机主要应解决当出现“无数据”的情况时,如何判断该状态是在任务内发生的还是在任务间发生的.为解决该问题,引入一个 flag 标志,当  $\text{flag}=1$  时,表示“任务间无数据”,当前任务完成;当  $\text{flag}=0$  时,表示“任务内无数据”当前任务未完成.问题转化为如何正确的 set/clear flag.考虑到  $t_2$  的产生是手动控制应用程序的时间,而  $t_1$  是 PC 机填充驱动缓冲的时间,显然  $t_2 \gg t_1$ ,设定一个时间基准  $t_s$ ,使  $t_1 < t_s < t_2$ .当出现“无数据”情况时,开始计时,若计时时间  $t > t_s$ ,表明这是“任务间无数据”,需要 set flag.系统复位时 set flag=1,当 flag=1 后,如果“有数据”条件出现,说明下一个任务开始了,因此,需要在正常发送完数据后 clear flag 即可.

各状态间跳转条件如下:

1.  $\sim \text{clr\_n}$
2.  $\text{clr\_n} \ \&\& \ (\text{watch\_dog} > \text{DELAY\_COUNT})$
3.  $\text{clr\_n} \ \&\& \ (\text{watch\_dog} \leq \text{DELAY\_COUNT})$
4.  $\sim \text{clr\_n}$
5.  $\text{clr\_n} \ \&\& \ \text{usb\_ef\_n}$
6.  $\text{clr\_n} \ \&\& \ (\sim \text{usb\_ef\_n}) \ \&\& \ \text{mission\_start}$
7.  $\text{clr\_n} \ \&\& \ (\sim \text{usb\_ef\_n}) \ \&\& \ (\sim \text{mission\_start})$
8.  $\text{cnt\_rd} \neq 37$
9.  $\text{cnt\_rd} == 37$
10.  $\text{cnt\_rd} \neq 37$
11.  $\text{cnt\_rd} == 37$
12.  $\text{cnt\_int} \neq \text{INT\_TIME}$
13.  $\text{cnt\_int} == \text{INT\_TIME}$

各状态跳转关系如图 4 所示.

### 3 实验结果

为了测试数据发送系统是否会出现数据缺失或时序错位等问题,特别构造一组有规律的数据用于系统测试.采用从零开始连续递增达到某一定值(需小于 FFFF 值)再连续减小到零,如此反复.通过将这样一组数据送入数据处理系统后输出的数据画图观察锯齿形图样就可以直观的发现是否存在数据缺失或者重复.通过使用这样的 1 组长数据,通过多次实验,出现了有一定规律的数据丢失现象,即接收到的数据比原数据文件少了其中的一些数值.经过反复试验和检查,最终确定了原系统 DSP 存在收到中断后间歇

性不相应的问题.通过修改 DSP 软件,消除了此现象.当连接传感器组进行实验时,因为采集的是模拟量,无法验证数据是否全部接收,此问题难以暴露.

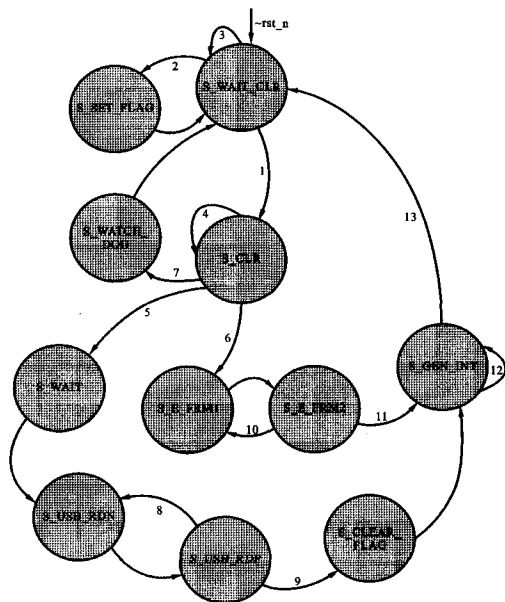


图 4 状态机结构图

### 4 结束语

本数据发送系统成本低廉,使用方便.若将 USB 接口部分与 FIFO 接口部分分开制作,可以通过更改 FIFO 接口的电路与时序配合多种不同的系统进行试验.相对于软件模拟,本系统不但可以用于进行算法验证,还可以验证硬件系统的可靠性与处理能力.可以重复载入外场实验的真实数据,节约实验成本.

### 参考文献:

- [1] 钱峰. EZ-USB FX2 单片机原理、编程及应用[M]. 北京:北京航空航天大学出版社,2006.
- [2] 陈文字. 有限自动机理论[M]. 北京:电子工业出版社,2007.
- [3] 王冠. Verilog HDL 与数字电路设计[M]. 北京:机械工业出版社,2006.
- [4] 唐俊. 基于 FPGA 和 DSP 的瞬态测量系统的设计[J]. 微电子学与计算机,2008,25(12):148-150.
- [5] 刘文怡,袁伟涛,张斌珍. 基于 FPGA 和 USB 的 64 路数据采集系统设计[J]. 微电子学与计算机,2008,25(12):151-153.

### 作者简介:

杨晓光 男,(1982-),博士研究生.研究方向为数字信号处理.