

无人机载设备状态监测系统的设计与实现

李文明^{1,2}, 张涛¹, 陈俊江¹

(1. 中国科学院 长春光学精密机械与物理研究所, 长春 130033; 2. 中国科学院 研究生院, 北京 100039)

摘 要: 论述了一种基于 SD 卡技术实现无人机载设备状态监测系统的实现方法。其核心为利用 SD 卡作为载体, 采用 AVR 单片机控制系统工作, 将机载设备关键的状态信息及监测结果实时存入 SD 卡中, 事后利用分析软件对 SD 卡中的数据进行分析处理, 并将处理结果以文件形式存入计算机中。描述了单片机与设备分系统 RS422 接口的硬件及软件设计方案, 与 SD 卡之间 SPI 接口的硬件及软件设计方案和分析软件的设计实现。系统在某机载项目中进行了实际的应用, 经过一年多的实践证明, 该系统有效地完成了无人机载光电仪器的状态监测与记录, 记录的数据为设备的维修与日常维护提供了重要依据。

关键词: 机载设备; SD 卡; 状态监测

中图分类号: TP274

文献标识码: A

文章编号: 1672 - 9870 (2008) 02 - 0017 - 04

Design and Realization of the state detection system for the device in pilotless aircraft

LI Wenming^{1,2}, ZHANG Tao¹, CHEN Junjiang¹

(1. Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy Of Sciences, Changchun 130033;
2. Graduate School of the Chinese Academy of Sciences, Beijing 100039)

Abstract: A method based on the SD card technology to implement the state detection of the device in pilotless aircraft is discussed. The core of this method is using the SD card to be a device carrier with the AVR SCM to control the whole system, which save the real-time key state information and detection result to the SD card, using analytic software to manage data stored in the SD card afterwards and saving the result to computer as format of a file. The blue print of interface between the SCM and slave system through RS422 bus is described. The blue print of hard ware and software of SPI interface between SD card as well as the analytic software are discussed. After more than one year of practice on some project, it is proved that the system can detect and memorize the state of photo electricity device on pilotless aircraft, which provide the gist for subsequent servicing and daily maintenance.

Key words: aircraft device; SD card; state detection

对无人机载设备的控制只能由地面站通过无线电链路的遥控通道来实现, 设备的工作也只能由地面站通过无线电链路的遥测通道进行监测并记录工作状态, 但是由于特定的任务要求超出无线电传输范围或者外界干扰等因素容易产生链路失锁, 设备的工作状态在地面站无法进行监控和记录, 这就要求机载设备能在工作的同时实时监测并记录自身的

工作状态, 以便事后分析。本文通过 AVR 单片机结合 SD 卡读取技术在设备工作的同时监测并记录设备的工作状态, 事后将记录的数据用分析软件进行分析处理并将处理结果保存为文件形式, 便于保存设备每次飞行的工作状态, 并为设备的日常维护与维修提供了保障。系统采用的 AVR 系列单片机, 运行速度快、结构简单, 具有高可靠性和良好的经

收稿日期: 2008-02-28

基金项目: 总参测绘局资助项目

作者简介: 李文明 (1978-), 男, 吉林农安人, 硕士研究生, 实习研究员, 主要从事航空侦查相机控制器研究。E-mail: snow_79@163.com

济性,能够满足系统的运行要求。SD卡的数据安全性、存储容量、性能表现以及使用环境也满足系统的使用要求。分析软件采用了 Visual C++ MFC 应用程序框架,融合读写文件技术,使软件简单实用。

系统整体框图如图 1 所示,整个系统分为状态监测记录和分析处理两个部分。其工作过程为:系统工作时通过 AVR 单片机的 RS-422 接口以总线方式将设备分系统的工作状态实时采集下来并将采集的结果实时写入 SD 卡中,事后通过计算机分析软件将 SD 卡中的数据读取出来并进行分析处理,将处理结果以文件形式存入计算机中。

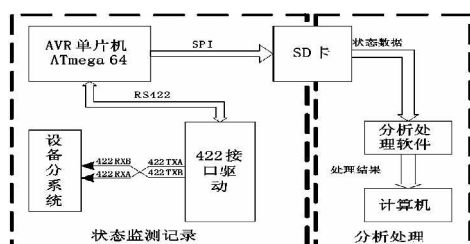


图 1 系统原理图

Fig.1 Schematic of system

1 监测记录系统

状态监测记录系统分为状态采集和记录两个部分。

1.1 状态采集系统

1.1.1 硬件设计

硬件电路原理框图如图 2 所示,设备与分系统之间采用 RS422 串行通信接口,RS422 接口抗干扰能力强,是一种单机发送、多机接收的单向、平衡传输规范。设备包括有几个不同的分系统,所以设备与分系统通信采用总线结构的主从式多机通信。输出接口芯片选用低功耗的差动线路驱动器 DS26C31,输入接口芯片选用接收器 DS26C32。

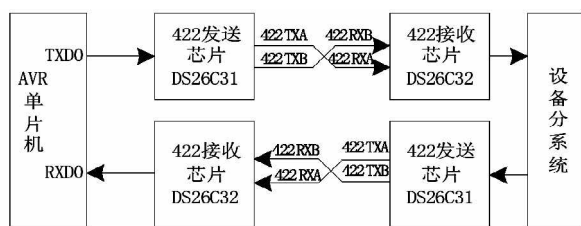


图 2 RS422 接口电路原理图

Fig.2 Structure Picture of Rs422 Interface

1.1.2 软件设计

采集数据接口软件与设备各分系统之间通信均采用查询方式,在每个通信周期内分别向每个分系

统发送请求信息并查询各分系统的返回数据,返回数据的第一个字节为分系统地址,如果地址正确则继续接收其它数据,否则放弃本帧数据,接收完毕后将分系统返回的状态信息存入寄存器中。

1.2 状态记录系统

1.2.1 硬件设计

硬件接口原理图如图 3 所示。状态记录系统采用 AVR 单片机的 SPI 接口与 SD 卡连接,其中单片机 ATmega64 是由 Atmel 公司研制开发的,具有双总线结构的 RISC(Reduced Instruction Set CPU),即精简指令集高速 8 位单片机^[3]。由于 SD 卡的接口电压为 3.3V,所以电源采用 TI 公司生产的 5V 转 3.3V 的电源转换芯片 TPS75533 来实现。根据系统运行速度以及产生串行通信的波特率的要求,选择 7.3728MHz 的晶振为单片机提供时钟信号。

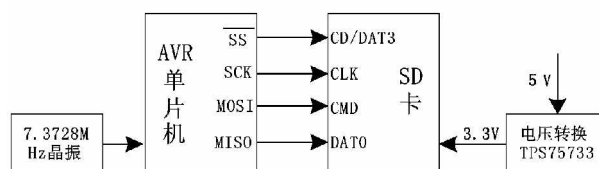


图 3 接口原理图

Fig.3 Schematic of interface

1.2.2 软件设计

软件设计主要是利用 AVR 单片机的 SPI 接口将采集到的状态数据实时写入 SD 卡中。

1.2.2.1 SD 卡初始化

通过与 SD 卡之间的握手信号来判定是否初始化正常。首先控制单片机的 IO 口设置 SD 卡有效,然后给定 100 个时钟循环等待 SD 卡载入,定义一个 6 个字节的命令字数组 CMD[6],首先写入复位命令 CMD[]={0x40, 0x00, 0x00, 0x00, 0x00, 0x95},然后发生该命令至 SD 卡,发送完毕后查询 SD 卡的响应状态字并设置计数器累加计数,如果计数器未满 50 且返回字节为 0x01,说明复位正常,如果计数器满则 SD 卡复位失败;在复位命令响应正确后,需要向 SD 卡发送初始化命令 CMD[]={0x41, 0x00, 0x00, 0x00, 0x00, 0xFF},与发送复位命令一样,等待返回数据,如果返回数据为 0x00,SD 卡初始化正常,否则失败。

1.2.2.2 配置 SPI 接口

ATmega64 单片机 SPI 接口通过软件配置 SPCR 和 SPSR 这两个寄存器来实现,具体配置如下:

$SPCR = SPCR | (1 \ll SPE) | (1 \ll MSTR) | (0 \ll$

$\ll \text{SPR0} \gg | (0 \ll \text{SPR1} \gg);$

$\text{SPSR} = \text{SPSR} | (1 \ll \text{SPI2X} \gg)$

SPCR 寄存器中 SPE 置 1 使能 SPI, MSTR 置 1 设置主机模式, SPR0 和 SPR1 设为 00 表示 SCK 频率为 $f_{\text{osc}}/2$ (f_{osc} 为系统晶振频率); SPSR 寄存器的 SPI2X 置 1 表示 SPI 的速度加倍, 可提高写入速度, 这两个寄存器其它数据位并未涉及, 不进行定义。

1.2.2.3 写 SD 卡

系统对 SD 卡的操作采用 SPI 模式^[1], 通过 SPI 消息中的命令包控制 SD 卡完成相应的操作, SPI 命令由 6 个字节 48 位数据组成。SD 卡满足 FAT 文件系统规范, 此文件系统由四个组成部分^[2]: 分区引导扇区、FAT 表、文件目录区和数据区。根据设计要求本系统对 SD 卡的操作主要是设置扇区长度、写文件名和写 FAT 链表和写数据。

设置扇区长度、写文件名和 FAT 表都是在 SD 卡初始化完成后进行的, 系统每次上电只需执行一次。发送 $\text{CMD}[] = \{0x50, 0x00, 0x00, 0x02, 0x00, 0xFF\}$ 命令来实现设置扇区的操作, 本系统设置扇区长度为 512 字节。写文件名和 FAT 链表是向文件名和 FAT 链表所在扇区分别写入预先规划好的数据, 都是对 SD 卡扇区进行写操作。扇区写操作相对复杂, 首先初始化 $\text{CMD}[] = \{0x58, 0x00, 0x00, 0x00, 0x00, 0xFF\}$, 然后根据需要写入扇区的地址确定 $\text{CMD}[1]$ 、 $\text{CMD}[2]$ 和 $\text{CMD}[3]$, 这三个字节根据下面表达式确定 (其中 addr 为扇区地址):

$\text{CMD}[1] = ((\text{addr} \& 0xFF000000) \gg 24)$

$\text{CMD}[2] = ((\text{addr} \& 0x00FF0000) \gg 16)$

$\text{CMD}[3] = ((\text{addr} \& 0x0000FF00) \gg 8)$

确定命令字并发送, 等待 SD 卡响应, 如果响应正确, 通过 SPI 接口向 SD 卡发送 100 个循环的周期时钟, 然后向写入起始字节 0xfe, 循环写入 512 字节数据, 写入 CRC 码 0xff, 以上操作完成后通过 SPI 接口监视 SD 卡的状态, 如果 SD 卡返回数据为 0x00 (默认为 0xff), 表示一次写入扇区操作完毕。

2 处理软件实现

SD 卡中的数据为分成多个文件进行记录, 每个文件记录大约 1h 的状态数据, 分析软件需要将 SD 卡中的数据依次读出, 并将数据进行分析处理并将数据进行保存, 待所有的数据处理完成后, 建立文件并将处理完成的数据以文件形式保存在计算机中。软件实现流程图如图 4 所示。

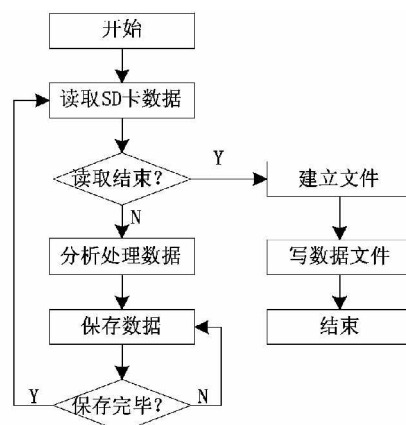


图4 软件流程图

Fig.4 Flow chart of software

软件采用 Visual C++ 6.0 基于 MFC 的对话框类创建, 分析处理部分在软件后台运行, 用户只需要简单的选择文件的读取路径和生成文件的保存路径, 点击分析转换按钮就可轻松实现分析处理功能。软件执行界面如图 5 所示。



图5 分析处理软件界面

Fig.5 Interface of analysis and process

图 5 的分析处理软件将 SD 卡中的源码数据处理成通用的 txt 格式文件, 可以用一些常用的处理软件对数据进行分析处理。下面的两组数据是从实验数据中截取的, 由于实验数据较大, 无法完全显示, 所以只截取部分数据供参考, 其中图 6 为源数据, 图 7 为处理后的数据。

```
00000000h:  F E4 61 D3 00 00 00 84 6E 00 07 80 55 08 DF 85
00000010h:  00 FA 9F E4 36 31 00 00 00 00 00 2C 66 00 84
00000020h:  0E 30 36 00 28 80 04 80 04 00 EF 00 A8 15 52 00
00000030h:  35 FD 5E 0F 3B 81 31 80 63 0E 30 36 01 18 9E 28
```

图6 源数据

Fig.6 Source data

焦 距	存储帧数	GPS时间	速 度	滚 动 角	俯 仰 角	偏航角	经 度	纬 度
110.66	0132	14:40:54	000	-00.20	-00.20	002.39	110.15506	35.38270
110.66	0132	14:40:54	000	-00.15	-00.25	002.24	110.15506	35.38273
110.66	0132	14:40:54	000	-00.05	-00.25	002.15	110.15506	35.38276
110.66	0132	14:40:54	000	000.00	-00.20	002.09	110.15506	35.38279
110.42	0132	14:40:54	000	-00.05	-00.15	002.06	110.15506	35.38282
110.66	0132	14:40:54	000	-00.10	-00.05	002.05	110.15506	35.38285

图7 转换后数据

Fig.7 Data of conversion

记录的设备工作状态对事后设备的维修以及出现问题的分析处理提供了重要依据, 图 8 为在设备调试过程中, 针对机载设备在进行电视跟踪过程中, 伺服控制系统经常发生飞车的问题对计算电视跟踪参数中的焦距数据进行分析的数据折线图。

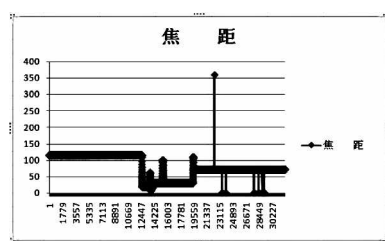


图8 焦距分析图

Fig.8 Analysis chart of focus

从图8中可以看出,焦距值在变化的过程中存在跳变,如果用跳变的数据来计算,脱靶量会产生一个异常的数据,这样伺服系统就会发生飞车现象。跳变的数据无规则,所以不能判定来源,考虑可能为干扰或通讯误码产生的,伺服系统根据记录的数据进行相应的野值剔除处理后,设备跟踪正常,从这个例子可以看出,本系统对设备的日常调试及维护起了重要作用。

3 结束语

系统使用的接口电路芯片都比较常见且价格低廉,使用 Kingston128MB 的 SD 卡作为存储介质,

可连续记录 20h 系统的工作状态,分析处理后的数据成功解决了设备在飞行中出现的各种问题以及设备自身存在的潜在问题。通过该系统在笔者参加的几个机载项目的实际应用证明,系统稳定可靠、实用性强,操作简单,为提高设备运行的可靠性以及可维修性提供了保障。

参考文献

- [1] SD Memory Card Specifications-PHYSICAL LAYER SPECIFICATION Version 1.01 April 15 2001[M/OL] 2007.3].<http://www.ouravr.com/bbs/>.
- [2] SD Memory Card Specifications-FILE SYSTEM SPECIFICATION Version 1.01 April 15 2001[M/OL] [2007.3].<http://www.ouravr.com/bbs/>.
- [3] 李长林. AVR 单片机应用设计[M]. 北京: 电子工业出版社, 2005.9.
- [4] 沈文, Eagle lee 詹卫前. AVR 单片机 C 语言开发入门指导[M]. 北京: 清华大学出版社, 2003.5.
- [5] 张克彦. AVR 单片机实用程序设计[M]. 北京: 北京航空航天大学出版社, 2004.2.
- [6] 何建辉. 实用 Visual C++ 6.0 教程[M]. 北京: 清华大学出版社, 2000.8.

(上接第 22 页)

4 结论

提出的异步多传感器跟踪融合方案在局部和全局两个层面上处理数据,解决了异步问题,在等待间隔 T 内的任何时间点都可以进行融合处理。所设计的滤波器,充分利用了多传感器信息充分的优势,又解决了多传感器常常不能同步的问题。今后研究可向具体算法方向考虑,算法的选择,对于整套跟踪系统性能的改进有着重要影响,可根据各传感器类型,跟踪目标类型等特性条件和系统特性等

条件设计适合相应跟踪系统的局部融合算法及全局融合算法。

参考文献

- [1] 何友, 王国宏, 陆大. 多传感器信息融合及应用[M]. 电子工业出版社, 2000.
- [2] 郑大钟. 线性系统理论[M]. 清华大学出版社, 2002.
- [3] Samuel Blackman, Robert Popoli. Design and Analysis of Modern Tracking Systems[M]. Artech House radar library 2000.
- [4] 周宏仁, 敬忠良, 王培德. 机动目标跟踪[M]. 北京: 国防工业出版社, 1991.