

基于 FPGA 和 ARM 的控制器设计及在快联锁保护系统中的应用

刘 鑫 蒋舸扬 沈立人 陈建锋

(中国科学院上海应用物理研究所 上海 201800)

摘要 联锁保护系统是上海光源为保护重要设备而设置的控制子系统, 快联锁保护系统是其中一部分, 其联锁响应时间要小于 1 ms。本文介绍了一种基于 FPGA 和 ARM 的快联锁保护系统的实现方案, 该方案利用 ARM 软件功能强及 FPGA 逻辑处理灵活和 IO 资源丰富的特点实现快联锁的功能。本文简要介绍了系统的设计方案和硬件结构, 及该控制器的测试结果。

关键词 快联锁保护系统, FPGA, ARM

中图分类号 TP277

上海光源(SSRF)的联锁保护系统是为保护重要设备而设置的控制子系统, 以在加速器各系统间建立确保设备运行安全的联锁逻辑系统^[1]。快联锁保护系统是其中一部分, 位于光束线前端接口处。光束线发生真空快泄漏事故时, 真空保护快阀须立即关闭, 以防止真空泄漏扩散。真空保护快阀不能承受同步光直接照射, 须真空泄漏信号给出 1 ms 内档掉储存环束流。目前, SSRF 联锁保护系统采用可编程逻辑控制器(PLC), 其时间响应达数十 ms, 无法满足快联锁的时间响应要求^[2]。

本文介绍一种结合 FPGA 硬逻辑高速信号处理能力和 ARM 数字功能扩展能力的快联锁系统硬件结构。其能克服 PLC 响应速度慢的缺点, FPGA 快速逻辑处理能力及稳定的性能可满足快联锁保护系统的需求。

1 快联锁控制器的硬件

1.1 总体要求

快联锁控制器的结构框图如图 1。快联锁保护系统输入各相关子系统的状态, 每个子系统运行正常与否都用一个数字量(STATUS 信号)表示。根据安全联锁系统失效安全的要求, 子系统运行正常时, 对应的 STATUS 信号为高电平; 子系统运行不正常时, 对应的信号就为低电平。所有状态信号构成 STATUS[N:0]并输入 FPGA 中进行逻辑处理, 得到锁存(LOCKED)信号。

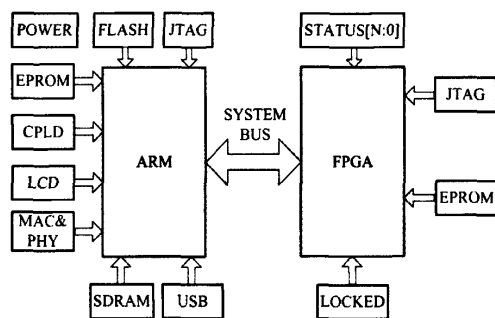


图 1 系统结构框图

Fig.1 System architecture.

当所有子系统运行正常时, LOCKED 管脚稳定地输出一个 TTL 高电平(逻辑数字量 1)。一旦某子系统发生故障, 经 FPGA 处理后 LOCKED 管脚会输出 TTL 低电平(逻辑数字量 0)来关闭高频功率输出, 档掉束流, 起到联锁保护作用。当故障排除后 LOCKED 管脚才输出高电平, 允许重新注入束流, 同时 ARM 通过总线接口(数据线、地址线)读取输入 FPGA 中的 STATUS[N:0]信号, 并通过其网口用 TCP/IP 协议将这些信号传输到操作面板接口(OPI)。

1.2 硬件结构

本文采取 ARM 和 FPGA 为核心的实现方案。FPGA 完成联锁逻辑信号的处理, 型号是 Altera EP1C6Q240。该芯片供用户使用的管脚达 185 个, 器件的逻辑处理响应速度可达 ns 级。可实现稳定可

靠的硬件联锁逻辑处理, 联锁响应速度快, 有灵活的逻辑再编程能力。

ARM 处理器选用 XSCALE PXA270 运行嵌入式操作系统, 实现联锁逻辑信号的读取及传输。该处理器最高主频达 624 MHz, 有丰富的外设, 并支持众多公司的嵌入式操作系统和开发系统, 如 Linux^[3]。

电源的稳定性是整个系统能稳定工作的关键。PXA270 工作核心电压为 1.4 V, 内存模块、PLL 模块和 IO 模块的工作电压分别为 1.8 V、1.3 V 和 3 V, LCD 显示屏的供电电压是 12 V, 其余模块的为 3 V。FPGA 工作电压是 3.3 和 1.5 V, 分别供给 FPGA 的 IO 模块和内部模块, 整个电源系统较复杂。为尽可能少的减少外接电源的电压总数, 从外部引进一个 5 V 和 12 V 电源, 其中 12 V 电压供给 LCD 显示屏, 5 V 电压通过器件转换为系统工作需要的各种电压。电源转换芯片是 LP3971 和 LT1764, 其电压和电流参数范围能满足系统需求, 其中 LP3971 和 LT1764 分别将 5 V 外接电源转化成 PXA270 和 FPGA 中各模块所需电压。

PXA270 内部无集成网络模块, 需外接一片 PHY+MAC 芯片来实现网络功能。我们用 LAN9111C 作 PHY+MAC 层设备, LAN9111C 所需时钟由 PXA270 提供, 该芯片可将 ARM 处理器的数据实时传输给远程操作界面上。PXA270 支持各类内存芯片, 控制器用 2 片 32 MB 的 Infineon HYB25L256160AC 构成系统内存, 总容量 64 MB。PXA270 启动时, 需对其内存存储器接口参数进行配置, 包括位宽、地址线的地址映射、内存工作的时钟频率等, 全部储存在系统启动 EEPROM 里。我们选用 ATMEL 公司的 AT24C01 作为启动 EEPROM, 系统上电后对系统进行最基本的配置和程序引导, 导入初始的启动代码。EEPROM 容量有限, 只能存储基本的配置程序, 不能提供足够空间存放操作系统映像、程序代码、常量表及一些系统掉电后保存的用户数据等, 故需其它存储介质。PXA270 外部存储器总线接口支持各种存储器芯片, 我们选用 2 片 Intel 公司的 StrataFlash Embedded Memory(P30)作为系统应用程序的存储介质, 总容量 64 MB^[4]。

1.3 ARM 与 FPGA 的接口设计

结构中可将 FPGA 视作 ARM 的一个外设, 根据 PXA270 处理器的可编程静态存储系统体系结

构, 将 FPGA 的空间定位于 n CS<4>管脚选择 0x1000_0000 到 0x1400_0000 共 64 MB 静态地址空间上。

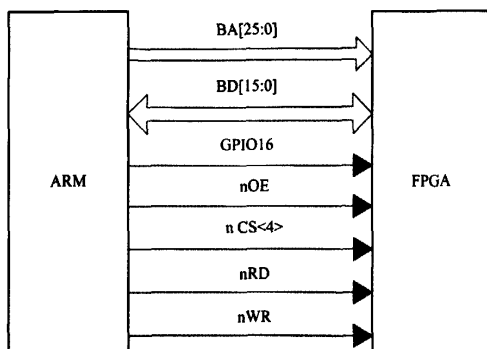


图 2 ARM 与 FPGA 的接口

Fig.2 Interface of ARM&FPGA.

如图 2, 该总线包括 ARM 芯片的地址总线 BA[25:0]、数据总线 BD[15:0]、控制总线及中断信号线(GPIO16), 其中控制总线包括使能信号(nOE)、片选信号(n CS<4>)、读信号(nRD)和写信号(nWR)。这样将 FPGA 芯片存储器化, 即 ARM 通过访问特定地址来控制 FPGA 工作。同时在 ARM 控制器端用定时中断查询方式判断 FPGA 状态。定时中断程序通过对 ARM 地址总线在 FPGA 中译码而顺序锁定各信号的电平值, 然后再经数据总线传回 ARM 并送至操作面板接口。

2 快联锁控制器的相关软件

系统软件主要包括嵌入式操作系统的移植、FPGA 联锁逻辑及联锁状态实时显示。联锁状态显示软件通过 PXA270 数据线及地址线读取 FPGA 联锁信号并传递至上层界面进行实时显示。

2.1 操作系统的运行及 FPGA 初始化

ARM 处理器支持众多公司的嵌入式操作系统和开发系统。本设计中裁剪并移植 Linux 2.4.21 内核, 可支持各类设备驱动, 如网络设备驱动、USB 主机驱动、串口驱动、键盘驱动及 ARM 对 FPGA 的驱动程序等。ARM 对 FPGA 的驱动程序是通过 fpga_init()函数初始化 FPGA 设备, 最终通过 init_module(fpga_init)在内核启动时初始化 FPGA 设备^[5]。fpga_init()函数的流程如图 3。

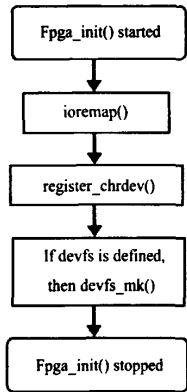


图 3 Fpga_init()流程图
Fig.3 Fpga_init() flowchart.

2.2 系统联锁功能的模拟

测试控制器的联锁逻辑用 Verilog HDL 硬件描述语言实现，通过 FPGA 编程软件将编写好的测试逻辑写入 FPGA 中，测试软件可读取 FPGA 的状态并将各联锁逻辑信号显示在操作界面上。

我们以 FPGA 扩展板上 8 个拨码开关来模拟快联锁系统的联锁输入信号，这 8 个拨码开关分别对应 FPGA 的 8 个联锁输入引脚，快联锁的输出信号由 FPGA 输出引脚对应的 8 个 LED 灯表示。拨动 FPGA 扩展板上 8 个拨码开关观察 8 个 LED 灯的状态，测试联锁逻辑及状态显示。拨码开关和 LED 灯的状态分别由界面 Switch Board 和 Led 栏显示，拨码开关向上拨，对应 Switch Board 栏的滚动条位置将如图 4，表示逻辑信号 1；反之表示逻辑信号 0。同时对应的 Led 指示灯亮，反之指示灯灭。

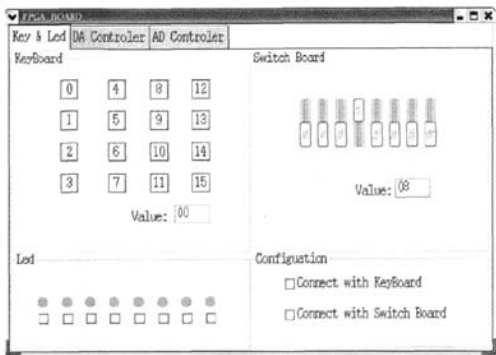


图 4 ARM 与 FPGA 的操作界面
Fig.4 Operation interface of ARM&FPGA.

用拨码开关对系统进行测试，界面中状态可正确反映联锁测试逻辑的执行。同时用 ALTERA 的 QuartusII 软件对快联锁逻辑的响应时间进行时序仿真，如图 5。

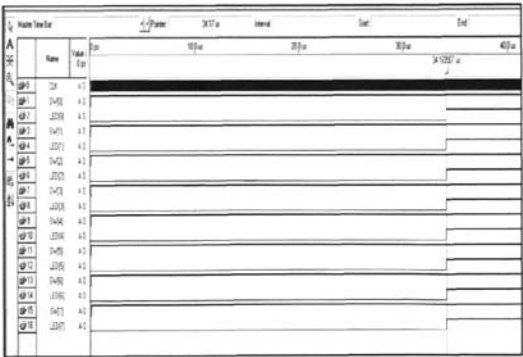


图 5 快联锁的响应时间图
Fig.5 Interlock response time.

CLK 为 25 MHz 时钟信号，SW[7..0]为输入信号，LED[7..0]为输出信号。在 CLK 第一个时钟周期内，SW[7..0]各信号分别产生一个从 0 到 1 的跳变，经逻辑处理后对应的 LED[7..0]也发生相应变化，其逻辑处理时间为 34 μ s，达到数十微秒级，小于系统要求的 1 ms，可满足快联锁的时间响应要求。由此，快联锁控制器的功能基本完成。

3 结论

本文设计并开发了一种基于 ARM 和 FPGA 的快联锁系统控制器。此控制器在一片 FPGA 芯片上实现联锁逻辑信号的处理，并以嵌入式 Linux 为操作平台在 ARM 处理器上完成对 FPGA 联锁状态的显示及人机交互接口等。与传统 PLC 实现的快联锁系统相比，有逻辑功能灵活、响应速度快及扩展性好等诸多优点。经初步实验验证，该控制器可满足快联锁系统的要求。由于此架构有较好的通用性及灵活性，稍加修改扩展就可应用到其它控制系统中去，有一定实用价值。

参考文献

- 1 上海光源初步设计报告
The preliminary design report of Shanghai Synchrotron Radiation Facility
- 2 上海光源储存环 MPS 快联锁控制机箱技术要求
The technical requirements of Machine protection Interlock System chassis based on Shanghai Synchrotron Radiation Facility
- 3 Intel PXA27x Processor Family Developer Manual, 2004
- 4 Howard Johnson. 高速数字设计. 北京: 电子工业出版社, 2003
Howard Johnson. High-speed digital design. Beijing: Electronic Industry Press, 2003

5 张 石. ARM 嵌入式系统教程. 北京: 机械工业出版社, 2008

ZHANG Shi. ARM embedded system tutorial. Beijing: Mechanical Industry Press, 2008

Design of a controller based on FPGA & ARM and its application in the SSRF fast interlock system

LIU Xin JIANG Geyang SHEN Liren CHEN Jianfeng

(Shanghai Institute of Applied Physics, Chinese Academy of Sciences, Shanghai 201800, China)

Abstract The interlock system is a control subsystem designed to protect essential equipment of Shanghai Synchrotron Radiation Facility (SSRF). As a part of the interlock system, the fast interlock system based on FPGA and ARM acts within 1 ms after a vacuum leak signal is given. In this paper, we describe the implementation of the fast interlock system. The design makes use of strong software capacity of the ARM, flexible logic process capability of FPGA, and rich IO resources to achieve the response time requirements of the system. The system solutions, hardware architecture and the test result of the controller are given, too.

Key words Fast interlock system, FPGA, ARM

CLC TP277