

# 微波开关适配器模块及其驱动程序设计

黄国兵<sup>1</sup>, 范云菲<sup>2</sup>

(1. 西安工程大学 计算机科学学院, 西安 710048;

2. 西安工程大学 电子信息学院, 西安 710048)

**摘要:** 根据低轨卫星通信网程控交换系统研究和产品开发的需求, 需要为一种遵循 PXI 总线标准的微波程控交换机设计基于微波开关的适配器模块, 并为该适配器模块移植设计专用的设备驱动程序; 微波开关适配器模块使用 PCI9052 总线桥电路实现了 PXI 总线接入功能, 并采用国产 FPGA 电路编程实现了适配器的接口逻辑; 在设计设备驱动程序时, 针对微波开关访问所要求的同步和互斥需求, 使用操作系统支持的事件对象、定时器对象和 DPC 例程等机制实现了对微波开关访问的安全操作管理; 项目开发完成之后, 采用工具软件对所设计微波开关适配器模块及其驱动程序的运行情况 and 操作功能进行了验证测试, 与预期目标相符。

**关键词:** 星链; 程控交换机; 微波开关; PXI 总线; FPGA; 设备驱动程序

## Design of Microwave Switch Adaptor Module and its Device Driver

HUANG Guobing<sup>1</sup>, FAN Yunfei<sup>2</sup>

(1. Computer Science College, Xi'an Polytechnic University, Xi'an 710048, China;

2. School of Electronics Information, Xi'an Polytechnic University, Xi'an 710048, China)

**Abstract:** According to the research and development needs of private branch exchange (PBX) systems for low-orbit satellite communication networks, it is necessary to design an adapter module based on microwave switch that meets to the PXI bus standard, along with a specialized device driver for this module. In the adapter module, a PCI9052 bus bridge circuit is used to achieve the PXI bus's access functionality, and a domestically produced FPGA circuit to implement the adapter's interface logic. When designing the device driver, mechanisms such as Event object, Timer object, and DPC routine supported by the operating system are utilized to meet synchronization and mutual exclusion requirements, thereby achieving safe operation management of microwave switch access. After the project development is completed, the tool software is used to verify the operation process and function of the designed microwave switch adaptor module and its device driver, which meets the expected goals.

**Keywords:** star link; PBX; microwave switch; PXI bus; FPGA; device driver

## 0 引言

与光纤通信信道一样, 微波无线通信信道也是一种重要的数据传输通路。以低轨卫星作为微波信号传输的中转站不仅能避开地表的物理障碍, 还能实现地表区域良好的覆盖。基于低轨卫星构建的微波通信链路(即星链)具有良好的发展前景<sup>[1-2]</sup>。随着美国 SpaceX 公司的 Starlink 星链系统投入商业化运营, 并在俄乌战场投入实际运用, 体现了低轨道卫星通信网络重要的战略价值<sup>[3-5]</sup>。低轨道卫星通信系统不仅在 5G/6G 无线通信网

络中有重要的作用, 还在空天预警等军事领域都有非常好的应用前景<sup>[6-8]</sup>。为了在这一战略领域实现对美国的追赶超越, 中国民用航天部门提出了中国星网星座、千帆星座和鸿鹄三号星座等 3 个低轨卫星通信网络的建设计划, 并开始付诸实施<sup>[9-10]</sup>。根据项目研究与试验的需求, 需要设计实现一款用于微波通信链路自动搭建/拆除功能的程控交换机系统。再结合用户需求规范书的要求, 项目组决定采用国产微波开关设计一块遵循 PXI 总线标准的专用适配器模块<sup>[11]</sup>, 搭建可以实现对众多微波开关组成的开关阵列进行程序控制的仪用计算机系

收稿日期: 2025-10-15; 修回日期: 2025-11-24。

基金项目: 陕西省重点研发计划项目(2022GY-058); 西安市科技创新人才服务企业项目(2020KJRC0022)。

作者简介: 黄国兵(1965-), 男, 硕士, 高级工程师。

通讯作者: 范云菲(2004-), 女, 在读本科生。

引用格式: 黄国兵, 范云菲. 微波开关适配器模块及其驱动程序设计[J]. 计算机测量与控制, 2026, 34(5): 206-213.

统实现所需的功能<sup>[12-14]</sup>。为此, 项目组先期已经开展了预研工作, 并完成了交换机原型系统的开发。在对所开发的原型机系统进行批产评审时, 发现其采用的 CPLD 电路已停产, 要求采用国产化元件予以替换, 并对原型机系统进行优化设计。由于微波程控交换机采用基于 PXI 标准的仪表计算机系统实现, PXI 总线和 PCI 总线的电气特性基本相同, 因此本文设计的适配器模块本质上就是一种基于 PCI 总线的 I/O 接口板; 又因该系统以 x86 处理器作为系统控制器, 以 Windows 作为操作系统, 本文所设计的设备驱动程序就是一种基于 Windows 的即插即用 (PnP) 型设备驱动程序 (Driver)<sup>[15-16]</sup>。这里主要就该适配器模块优化和设备驱动程序移植的实现技术进行具体介绍。

### 1 基于微波开关的 I/O 适配器模板设计

通过分析基于 PXI 总线的微波程控交换系统的技术规范, 每台 19 英寸程控交换机机箱可以插接 15~17 块微波开关适配器模块, 每个模板又可以装配 2~6 个微波开关, 这样每台程控交换机就能实现对多达几十个微波开关构成的开关阵列进行分/合操作, 实现众多从呼叫方到被叫方之间的微波通信链路的自动搭接/拆除的功能。遵循 PXI 总线通用模块标准的微波开关适配器模块插接在 PXI 总线机箱 (本文采用 3U 尺寸规范) 中运行<sup>[11,17]</sup>。

根据项目需求, 本文设计的微波开关适配器模块需要接入两种类型的微波开关, 分别是 SPDT 型和 SP6T 型微波开关。考虑到 PXI 总线模板的结构和微波开关的外形结构, 微波开关与适配器模板采用一体化方式安装, 每个 SPDT 型模板可以安装 2/4/6 个微波开关, 分别占用 1/2/3 个 PXI 总线槽位, 每个 SP6T 型模板只可以安装 2 个微波开关, 占 2 个槽位。通过分析所使用的

微波开关的技术手册, 两种型号微波开关的微波信号的工作频率都是 50 GHz, 都带有遥测信号<sup>[16]</sup>, 都采用 +3.3~5 V TTL 信号驱动操作, SPDT 型开关采用电保持工作方式, SP6T 型开关采用磁保持工作方式。每个 SPDT 开关的控制需要两路 DO (Digital Output) 控制信号, 分别是 E [1..2], 每个 SP6T 开关的控制需要 6 路 DO 信号, 分别是 E [1..6], 每个 SP6T 开关还需要 1 个复归信号 RST。此外, 每个 SPDT 开关状态的遥测信号检测需要两路 DI (Digital Input) 信号, 分别是 C [1..2], 每个 SP6T 开关的检测需要 6 路 DI 信号, 分别是 C [1..6]。

#### 1.1 微波开关适配器模板的总体解决方案

为了完成一款基于 PXI 总线的 I/O 适配器模板设计, 需要完成其与 CPU 之间实现数据传送的 PXI 总线接口, 与外部设备 (这里为微波开关) 之间数据传送的外设接口, 以及实现对外部设备进行访问控制的 I/O 适配器 3 个部分电路的设计工作, 模块的总体解决方案如图 1 所示。

为了实现适配器模块到 PXI 总线的接口, 本文选择了技术成熟且价格低廉的 PCI9052 总线桥电路实现这一功能, 该芯片符合 PCI r2.1 协议规范, 支持 132 MB/s 的突发传输, 支持本地 8/16/32 位数据总线宽度, 满足本项目的需求<sup>[18]</sup>。

其次, 针对对微波开关访问控制的适配器电路, 由于每个模块最多支持访问控制 6 个 SPDT 型或者 2 个 SP6T 型微波开关, 6 个 SPDT 型微波开关驱动信号为 S [1..6] E [1..2], SiE1 执行第 *i* 个开关的合操作, SiE2 执行第 *i* 个开关的分操作; 6 个 SPDT 型微波开关的遥测状态信号为 S [1..6] C [1..2], SiC1 为第 *i* 个开关的合状态遥测信号, SiC2 为第 *i* 个开关的分状态遥

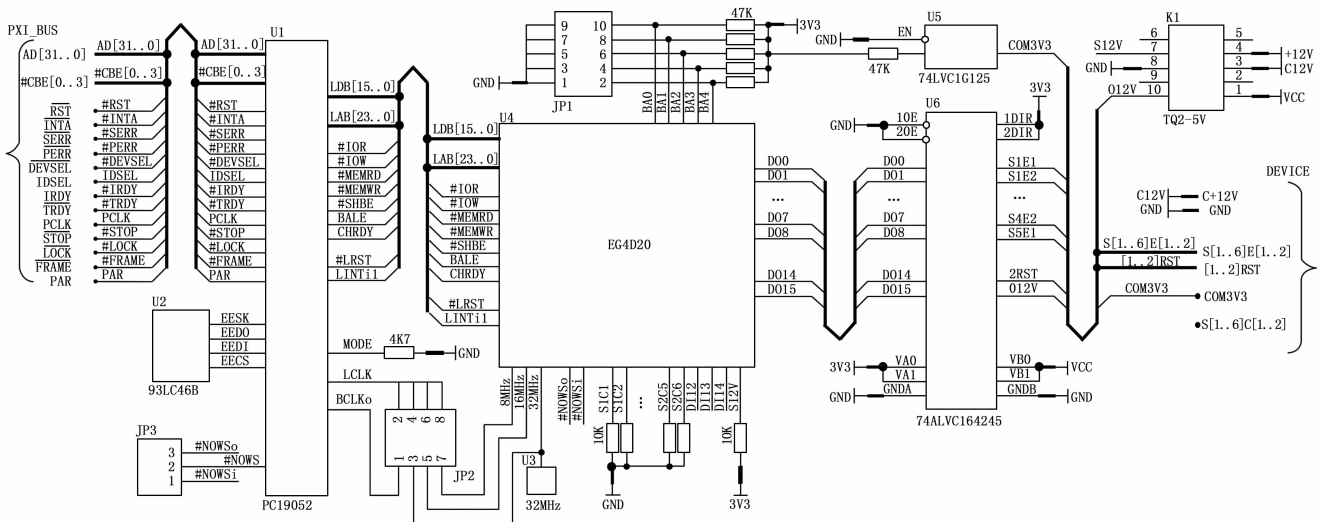


图 1 微波开关适配器模板的总体解决方案

测信号。2 个 SP6T 型微波开关的驱动信号为 S [1..2] E [1..6], SiEj 为第 i 个开关的第 j 个触点的合操作, [1..2] RST 为第 i 个开关的复位操作; 2 个 SP6T 型微波开关的遥测信号为 S [1..2] C [1..6], SiCj 为第 i 个开关的第 j 个触点的遥测信号。由此可见 6 个 SPDT 微波开关控制需要 12 路 DO 信号驱动, 2 个 SP6T 微波开关驱动需要 14 路 DO 信号驱动; 6 个 SPDT 和 2 个 SP6T 微波开关的遥测信号都只需要 12 路 DI 通道进行采集, 因此, 适配器模块采用 1 个 16 位读写双向寄存器端口即可完成对所需的所有微波开关进行访问控制的要求。

在微波开关驱动电路部分, 由于微波开关与适配器底板一体化安装, 且微波开关内部线圈驱动和遥测状态信号指示都采取了光电隔离的措施, 对微波开关的操作驱动采用一片 74ALVC164245DGG 16 位双向电平转换收发器芯片实现, 该电路可以产生 16 个分/合操作信号, 在适配器电路侧采用 3 V 信号电平, 在微波开关侧则使用 5 V 信号电平, 以提高信号驱动的能力。对于微波开关的遥测信号采集, 由于其内部采用共阳极方式实现开关状态指示, 这里采用 74LVC1G125 提供阳极的 3 V 公共信号, 对每个信号下拉 10 k 电阻之后直接接入 FPGA 电路的 I/O 引脚, 采用零握手线联络方式即可直接读取其指示的遥测状态。图 1 中指定本地总线的运行方式 MODE 信号下拉接地, 确定使用非复用方式访问, 适配器的 I/O 端口地址由 JP1 跳线器设置。

微波开关适配器在本地采用 16 位 ISA 总线接入 PCI9052 的本地总线接口。由于 PCI9052 电路对本地 ISA 总线有专门支持, 适配器电路只需要完成本地总线时钟 LCLK, ISA 总线中断源信号合并, 本地信号读写是否插入等待的控制信号 # Nows 等逻辑电路设计即可。

此外, 考虑到微波开关在加电启动过程中因 +12 V 工作电源不稳定会导致开关触点发生抖动的问题, 这里采用系统启动后延迟 10 s 再给微波开关供电的方法解决这一问题。+12 V 电源的工作状态 C12V 利用适配器端口寄存器的第 15 位做指示。

## 1.2 微波开关适配器电路设计

根据项目批产评审时提出的核心元件国产化要求, 本文选用安路 EG4 系列 FPGA 中的 EG4D20 完成微波开关适配器电路的设计<sup>[19]</sup>。EG4D20 具有丰富的 I/O 资源和强大的逻辑处理能力, 并支持用户设计 IP 保护和位流加密保密等功能, 集成了一块 8 M \* 16 位的 SDRAM, 满足本期项目设计和进一步演进设计的需求。

通过上一小节的分析, 微波开关适配器访问控制电路的核心是对 1 个 16 位读写双向寄存器端口的操作。

根据应用需求, 微波开关分为 SPDT 和 SP6T 两种类型, 每种类型又有电保持和磁保持两种工作方式。对于电保持开关, 开关触点状态由端口输出的电平状态确定, 当端口输出高电平, 开关触点闭合, 反之亦反; 对于磁保持开关, 开关触点闭合由端口输出的脉冲信号触发, 一旦触发, 开关触点就一直保持, 只有当该开关被 RST 信号复归, 开关触点才被断开。对微波开关控制操作时, 端口输出电平必须保持 15 ms 其触点的状态才能稳定。针对以上操作规则, 适配器寄存器端口的访问按如下 3 个步骤执行:

1) 扫描系统控制器对本端口寄存器的 outp 操作, 将数据总线的的数据传送至寄存器 now\_LDB [15: 0] 保存, 采用 HDL 编程实现的逻辑代码如下。代码中 IOW 和 PCSn 分别是 I/O 写信号和端口的片选信号。

```
always @(posedge IOW) begin
    if(! PCSn) begin
        now_LDB <= LDB;
    end
end
```

2) 检测新输入的数据是否有变化, 如果有变化, 说明有对微波开关的控制操作, 设置控制有效的标记 CTRL\_valid=1。为了记忆先前的状态, 定义了寄存器 old\_LDB [15: 0] 来缓存上一次执行过的老状态。HDL 代码如下。代码中 POWER\_EN=1 表示微波开关适配器已正常启动; SYS\_CLK 为控制行为的节拍间隔, 微波开关适配器本地的 SYS\_CLK 频率为 32 MHz。

```
always @(posedge SYS_CLK)
if(! POWER_EN) begin //适配器未启动
    old_LDB <= now_LDB; //无效操作
    CTRL_valid <= 1'b0;
end else begin //适配器已正常启动
    if(old_LDB != now_LDB) begin
        old_LDB <= now_LDB;
        CTRL_valid <= 1'b1; //有操作事件
    end else begin
        old_LDB <= now_LDB;
        CTRL_valid <= 1'b0; //无操作事件
    end
end
```

3) 根据系统控制器下达的控制命令, 控制指定的微波开关动作。考虑到需要控制的微波开关分为 2 种类型, 每种类型的开关又有 2 种工作方式, 控制指定的微波开关动作的过程方法不同, 为此, 本文设计了一个简单的 3 状态的状态机实现所需要的控制操作, 如图 2 所示。

实现图 2 所示状态机的逻辑代码如下。代码中, 为

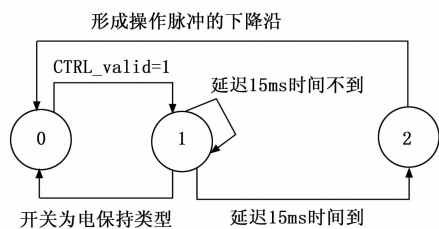


图2 微波开关访问操作的状态机

了便于理解,采用了中文注释协助表达;为了节省篇幅,对连续赋值语句采用了省略号替代。微波开关控制行为也按 SYS\_CLK 节拍步进运行。状态机的状态变量为 state, 延时定时器变量为 delayTI [31: 0], 15 ms 的计数值为 480 000。状态机的 3 个状态 0/1/2 使用 case 结构进行分别进入分支处理。在状态 0, 如果 CTRL\_valid=1, 则进入状态 1。在状态 1, 首先将系统控制器下达的指令传送至输出信号管脚, 再针对微波开关的工作方式分别提供电平和脉冲控制信号。如果是电保持工作方式, 则本次操作在设置寄存器状态, 产生输出电平之后结束, 返回状态 0; 如果是磁保持工作方式, 则在形成持续时间为 15 ms 的高电平之后转入状态 2。在状态 2, 针对磁保持继电器脉冲触发的要求, 产生脉冲信号的下降沿之后转至状态 0, 结束本次操作。对微波开关同步和互斥访问的要求, 本状态机不支持, 由下节说明的软件部分实现。

```

always @(posedge SYS_CLK)
if(! POWER_EN) begin //适配器未启动
    delayTI <= 32'd0; //无效操作
    state <= 2'd0;
end else begin //适配器正常工作
    case(state) //检测状态机当前所处状态
2'd0: if(CTRL_valid) begin //状态 0
        state <= state + 3'd1; //到状态 1
    end
2'd1: begin //状态 1
S1E <= now_LDB[1:0]; //输入传至输出端口
..... //包括 S[2:6]E,RST
if(now_LDB[13] 1'b0) begin //电保持类型
    state <= 3'd0; //到状态 0
end else begin //磁保持类型
    if(delayTI < 32'd480000) begin //时间不到
        delayTI <= delayTI + 32'd1; //时间+1
        state <= state; //状态不变
    end else begin //时间到
        delayTI <= 32'd0; //时间清 0
        state <= state + 3'd1; //到状态 2
    end
    end
end
end
end
end

```

```

2'd2: begin //状态 0
    S1E <= 2'b00; //形成脉冲下降沿
    ..... //包括 S[2:6]E,RST
    state <= 3'd0; //到状态 0
    end
default: state <= 2'd0; //状态非法,状态归 0
endcase
end
end

```

4) 对微波开关状态的遥测信号采集, 采用连续赋值语句实现, 代码如下。代码中 pSTAT 为微波开关 +12 V 工作电源的状态。

```

assign LDI[1:0] = ((! PCSn) & (! IOR)) ? S1C : 2'bzz;
assign LDI[3:2] = ((! PCSn) & (! IOR)) ? S2C : 2'bzz;
..... // 包括 LDI[11:4]部分位的连续赋值
assign LDI[15] = ((! PCSn) & (! IOR)) ? pSTAT : 1'bz;

```

### 1.3 微波开关适配器的端口地址配置

微波开关适配器作为计算机系统的一个功能部件, 为了使用 outp 和 inp 命令对其 I/O 端口进行访问操作, 必须为所设计的寄存器端口编排访问地址。PCI9052 总线桥支持对 5 个本地地址空间进行访问操作, 分别是 LAS0~3 和 ROM 空间。微波开关适配器只使用了一个 16 位 I/O 读写双向端口, 因此只需要启用其中 LAS1 空间即可。考虑到 PCI9052 内部访问的需要, 本文也启用了 LAS0 空间, 其它本地空间禁用。

为了实现对本地空间的访问, PCI9052 为每个本地空间设计了基地址、范围、总线区域描述符 (Bus Region Descriptor, BRD) 和片选基地址等 4 个配置寄存器。根据应用需要和 PCI9052 的要求, 本文规划的 LAS0 空间的基地址为 0x1000, 区域大小为 1k 字节; LAS1 空间的基地址为 0x300, 区域大小为 32 字节 (16 个 16 位 short 字)。由此可得配置寄存器参数如下: ① LAS0 的地址范围为 0~0x3ff, LAS0RR 为 0xFFFF FC00。LAS1 的地址范围为 0~0x1f, 且 LAS1 定义为 I/O 空间, bit0 置 1, LAS1RR 为 0xFFFF FFE1。② LAS0 的基地址为 0x1000, bit0 置 1, LAS0BA 为 0x0000 1001。LAS1 的基地址为 0x0300, bit0 置 1, LAS1BA 为 0x0000 0301。③ LAS0 和 LAS1 都为 16 位 ISA 总线的访问操作, LASxBRD 均为 0x0040 0022。④ LAS0 和 LAS1 区域大小分别为  $2^{10}$  和  $2^5$ , CS0BASE 为 0x0000 1201, CS1BASE 为 0x0000 0311。在确定 PCI9052 的配置参数之后, 就可以使用 PLXMon 软件将这些参数输入系统并写入到如图 2 所示的 EEPROM。一旦将配置参数保存在 EEPROM, 适配器模块启动时就以这些参数为准确定 PCI 总线的工作方式和访问参数。

根据 PCI 总线的规范, 微波开关适配器的本地空间

要映射到 PCI 侧对应的 PCIBAR 空间来访问, LAS0 和 LAS1 分别映射到 PCIBAR<sub>2</sub> 和 PCIBAR<sub>3</sub>。系统启动时 BIOS 自动将本地 I/O 空间映射至 PCIBAR<sub>3</sub> 空间 (PCI 总线侧的 I/O 地址空间)。比如: 模板调试时, 跳接图 2 中 JP1 设置本地末位地址为 6, 加上 I/O 基地址 0x300 后得到本地地址为 0x306, 系统启动后 BIOS 将本地基址映射到 PCIBAR<sub>3</sub> 的基地址, 加上偏移后即得到 PCI 侧的端口地址为 0x3086, 微波开关适配器的驱动程序、PLXMon 软件等通过该地址就能对所设计的端口进行访问操作了。

## 2 微波开关适配器模块的驱动程序设计

根据项目需求, 基于 PXI 总线标准的微波程控交换机的系统控制器的核心为 32 位的 x86 CPU, 软件平台为 Windows 7 及更高版本, 本节就基于 Windows 的 PnP 型的设备驱动程序移植设计进行说明。

### 2.1 微波开关适配器模块驱动程序的架构

由于微波开关适配器基于 PCI9052 总线桥电路设计, 该电路自带的 RDK 工具提供了电路板硬件和驱动程序软件的 Demo 样板。通过分析 RDK 提供的 Demo Driver 的架构体系, I/O 模块的驱动程序由芯片级 (Chip) Driver、服务级 (Service, Svc) Driver 和 API 接口程序三部分组成。用户 App 为了调用驱动程序进行相应的 IRP (I/O Request Packet) 操作, 先调用 API 接口的函数, 该函数调用 Service 级 Driver 中的 IRP 派遣函数, Service 级 Driver 派遣函数则根据 Chip 型号调用指定 Chip 的 IRP 派遣函数实现相应的功能。

为了简化驱动程序的设计工作, 微波开关模板的驱动程序继承了 Demo Driver 的架构体系, 对其中不采用的部分予以裁剪, 并扩展微波开关访问需要的特定功能, 以完成本适配器模板驱动程序的移植任务。具体实现时, 为微波开关适配器驱动程序选用的 Chip 级 Driver 为 Plx9050 目录中的驱动程序 Plx9050.sys, Service 级抽象接口类的 Driver 为 PlxSvc.sys<sup>[20]</sup>, 以及对 IRP 的派遣函数的调用进行封装后的 API 接口函数库 PlxAPI.dll。

### 2.2 微波开关访问的同步与互斥机制

资源管理是操作系统的重要功能, 资源管理存在同步和互斥的问题, Windows 设计了 Spinlock, Event, Mutex, Semaphore 等机制实现资源的同步和互斥管理功能。就微波开关的访问控制而言, 其本身也是临界资源, 一旦被占用, 只有在该开关被占用者释放之后方可被其它用户使用。微波开关的遥测状态的检测则存在同步问题, 只有在控制操作稳定且产生可靠的遥测状态之后方可被读取。由于 1.2 节所实现的适配器访问逻辑只实现了按时序访问端口的功能, 并不支持微波开关的互

斥和同步访问的机制, 因此, 项目软件需要利用 OS 提供的同步和互斥机制实现对微波开关的安全操作管理。

在操作系统和底层驱动程序的支持下, 可以采用同步操作和异步操作两种方式对临界资源进行同步和互斥管理, 本文采用了其中的异步操作方式。在异步操作时, 应用 App 的 DeviceIoControl 函数不等待 IRP 结束, 而是让应用 App 直接挂起返回, 当 IRP 操作完成之后, IoCompleteRequest 函数通知 IRP 操作完成, 让等待该事件的应用 App 进入就绪状态。可见, 异步操作方式时 CPU 的使用效率更高。采用异步方式时, 应用 App 侧执行微波开关控制操作的流程如图 3 所示。

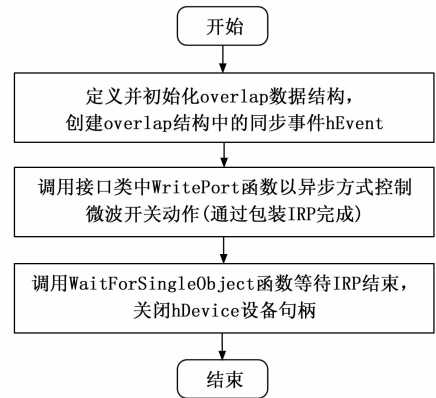


图 3 应用 App 采用异步方式进行开关操作的流程

此外, 按照微波开关数据手册的要求, 其每次进行分合控制操作之后, 需要间隔 32 ms 之后再进行一次操作, 以保证微波开关操作的稳定可靠性, 否则会造成开关触点的机械颤抖, 甚至会影响开关的寿命。

为了实现连续两次操作之间间隔 32 ms 的要求, 可以采用如下几种方案: 1) 应用软件对每个微波开关设置一个保护定时器, 每次操作时触发定时器并标识开关操作忙状态, 定时时间到达之后, 清除忙状态, 标记开关空闲; 每次操作前先检查开关的使用标记, 如果开关空闲, 方可进行控制操作。2) 在驱动程序中对操作对象进行保护, 延迟时间到达之后, 才释放该资源。3) 在电路板硬件电路设计时对开关操作进行延时保护。对比以上 3 种方案, 由于应用软件由第三方公司开发, 考虑到底层系统设计的透明性原则, 决定对微波开关操作的间隔延迟保护工作在驱动程序中实现, 以简化应用软件的设计。由于适配器逻辑负责实现每次操作的控制电平要持续 15 ms 的功能, 两次操作间隔 32 ms 的保护功能留给软件部分来实现, 以降低电路设计的复杂性, 为此操作间隔延迟保护的功能采用方案 2) 来实现。就开关操作间隔保护的本质而言, 也是物理对象操作的互斥与同步的问题: 即在对开关操作未完成之前不允许进行新的操作, 只有间隔时间到, 方可进行新的操作。由

于交换机系统基于多任务操作系统设计, 不能简单使用软件延迟等待的方法实现, 采用操作系统底层的延迟过程调用 (DPC, delay process call) 机制来实现更合适。操作系统的 DPC 例程与中断服务例程的实现方法类似, 在进行对象操作后, 将进行本次 IRP 操作的 App 挂起, 采用 KeSetTimer 内核函数设置并开启定时器计时, 并将该定时器和指定 DPC 例程的绑定。在设定的延迟时间到达时, 内核中的软中断机制就自动执行 OnTimerDpc 例程, 将被挂起的 IRP 的状态置为 STATUS\_COMPLETED, 并结束该 IRP 操作, 让等待该事件的应用 App 恢复就绪态。也就是说, 在应用程序发起对某个开关操作之后, 由于该开关被占用, 以致调用该操作的 App 被挂起, 只有延迟间隔到达之后调用该操作的 App 才能进入就绪态而进一步运行。使用这一方法的好处是: ①延迟等待期间应用 App 挂起, 不占用 CPU 时间, 从而提高了 CPU 的运行效率; ②通过 IRP 与操作系统的 hEvent 对象关联, 当定时间到, DPC 例程设置 IRP 结束, 图 3 所示的流程中的 WaitForSingleObject 函数等待的条件满足而进入就绪态。应用 App 程序按正常流程编写即可, 不需要增加其它代码, 即对应用 App 设计透明, 从而简化了应用 App 的设计。由此可见, 采用 DPC 例程实现操作间隔延迟保护功能时, 既不需要 CPU 死等而影响系统性能, 也不增加应用 App 设计的复杂性。OnTimerDpc 例程的流程图如图 4 所示。

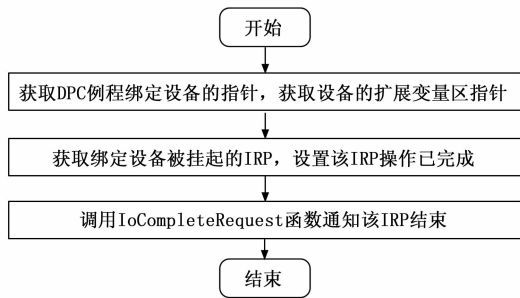


图 4 OnTimerDpc 例程的流程图

应用 App 设计时, 如果采用单线程方式实现对多个微波开关进行控制操作, 当应用软件对某个微波开关操作时, 一旦进程被挂起, 其它开关的操作就需要等待; 如果采用多线程方式编程, 即对每个微波开关创建一个线程进行操作时, 则可以达到多个开关并发操作的效果。在设计微波程控交换机系统的测试软件时, 由于不强调并发操作的效果, 可以采用单线程方式进行应用软件设计。对于现场在线运行的程控交换机系统, 上位决策计算机系统通过网络通信报文下达对某个微波开关操作的命令, 由于每次微波开关操作加上间隔等待的总时间为 47 ms, 约 0.05 s, 而上位计算机的对微波开关

的控制属于慢操作, 实时性要求不高, 因为一台 PXI 总线交换机系统最多可以对约 30 个微波开关进行操作, 采用单线程编程时, 需要等待的最长时间约为 1.5 s, 而实际一条通信链路建立之后平均持续时间远大于 1.5 s, 因此, 交换机系统的多任务软件采用单线程方式编程就可满足需求。如果微波程控交换机系统的在线运行软件的实时性不满足上位机系统对其并发性的要求, 则优化改进策略是采用多线程方式设计应用 App, 使任何微波开关的操作都不影响其它开关的及时操作, 而本文在驱动程序中采用 DPC 机制设计的间隔延时方法支持应用 App 的多线程方式编程。

### 3 实验结果与分析

在完成微波开关适配器模板制造和设备驱动程序的移植设计之后, 进一步对自带的 Demo INF 文件进行适应性修改, 并生成了 CatalogFile 签名认证文件 PxiSmc.Cat, 使得所设计的微波开关模板具备插接到交换机系统中正常运行的条件。

#### 3.1 使用 DebugView 观测和调试驱动程序

由于运行在内核模式下的驱动程序不能像用户 App 那样可以采用 IDE 集成开发环境单步跟踪的方法进行调试, 需要将程序运行的过程信息输出显示在屏幕上进行交互调试。在具体实现时, 驱动程序采用 DebugPrintf 和 InfoPrintf 两个宏向屏幕输出显示调试信息和运行信息, 使用 DebugView 软件既能监听应用层软件输出的 log 信息, 还能监听来自内核层驱动程序输出的 log 信息, 据此可以观察驱动程序内部指定代码运行的情况, 达到对驱动程序调试的目的。图 5 为测试时观测到的 PlxSvc 驱动程序输出的 Debug 信息的片段。

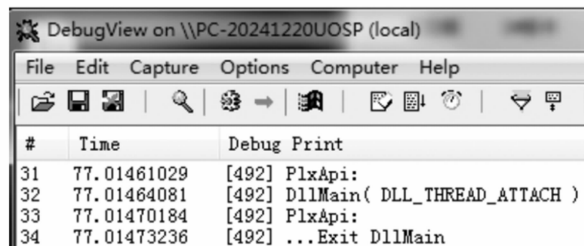


图 5 驱动程序提示的调试和运行信息

#### 3.2 使用 PLXMon 对 I/O 适配器进行操作访问

在交换机系统启动运行之后, 采用 PLXMon 工具软件通过设备驱动程序对微波开关模块的 I/O 适配器进行了访问操作, 以验证对微波开关操作控制和遥测状态校验功能的正确性, 进而验证适配器模块及其驱动程序的正确性。图 6 所示为在 PLXMon 软件的行命令区用监控命令进行的操作和操作之后回显的信息。其中 0x3086 为映射到 PCI 侧的 PCIBAR<sub>3</sub> 的端口地址, iw/ow 分别为 16 位端口的读/命令。

1	iw 3086	6	ow 3086 0002
2	7FFF	7	iw 3086
3	ow 3086 0001	8	7FFD
4	iw 3086	9	ow 3086 0011

图 6 用 iw/ow 命令对适配器进行访问操作的结果

图 6 中，行 1 为读 3 086 端口，行 2 为 iw 命令的明显信息，指示微波开关电路的 +12 V 电源的状态正常；行 3 为合 1# 开关命令，可以听到开关动作的响声，用万用表检测开关的输出通路，蜂鸣器响，说明开关合操作正确；行 4 为读开关的遥测状态，结果正确；继续进行访问操作，开关动作的结果和遥测状态读取都正确。上述测试结果也说明所设计的微波开关适配器模块及其驱动程序运行正常。

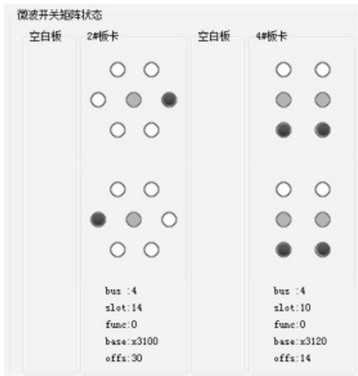
### 3.3 使用专门软件对微波程控交换机进行检测

为了方便对所设计的微波开关适配器模块以及所搭建的微波交换机系统进行生产调试和拷机测试，项目组基于 Windows 操作系统和所设计的设备驱动程序，采用 C#.net 设计实现了一款专门的测试软件。该软件通过图形界面实时显示交换机系统中插接的所有模板和模板上装配的开关的运行状态，以及执行测试过程的日志条目和关键参数，支持对指定板卡上的指定开关、指定板卡和系统中所有板卡检测任务，能够让测试人员观察到详细的测试过程和微波开关动作的实时状态和运行参数、测试结果，软件界面如图 7 所示。

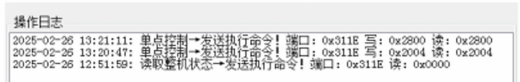
自产品定型生产以来，已完成 2 批次共计 5 台微波程控交换机系统的生产调试和出厂拷机测试，其中第 1 批次 2 台，每台 3 块双 SPDT 适配器卡和 3 块双 SP6T 适配器卡，第 2 批次 3 台，每台 2 块四 SPDT 适配器卡和 3 块双 SP6T 适配器卡，共计 27 块板卡，66 只微波开关。拷机测试时，被测设备每天上午、下午和晚上各 1 次对所有交换机系统进行一次整机中所有微波开关次第分合操作，记录测试结果。每个批次进行 168 小时（7 天）连续测试。根据产品整机拷机质量记录表的测试结果，已经进行的 189 台次交换机系统测试（对应 1 386 只微波开关控制测试）尚未出现操作失败的情况，表明所设计的模板和配套的设备驱动程序具有满足用户需求的稳定可靠性指标。

### 4 结束语

针对微波开关适配器模块优化和该模块的设备驱动程序移植设计的需求，首先采用一种国产的 FPGA 电路替代了原型机系统中的已停产的电路，并对原型机进行了有针对性的优化改进。其次，在借鉴 PCI 总线桥电路附带的 Demo Driver 的基础上，根据本项目的实际需求，先对其进行剪裁处理，删除其中 DMA 操作和 In-



(a) 开关模板实时状态显示区（局部）



(b) 操作日志条目显示区（局部）



(c) 操作面板区

图 7 微波交换机系统测试软件界面区域（局部）

errupt 操作等不使用的內容；再针对微波开关控制输出和遥测状态检测的访问规则要求，采用 Windows 内核提供的同步与互斥的机制实现了控制操作的异步互斥访问；并采用 DPC 定时器机制实现了微波开关连续两次操作间隔要大于 32 ms 的要求。在项目研究和产品设计开发伊始，考虑到首先要有可用于工程试验的产品，拟先采用“PCI9052 + CPLD”或“PCI9052 + FPGA”方案设计可用的产品，然后对 PXI 总线接口和适配器控制逻辑进行改进设计，进一步选用 FPGA 的 IP 核实现 PXI 总线接口，替代目前使用的 PCI9052 总线桥电路，再根据产品生产的批量决定是否采用 ASIC（Application Specific Integrated Circuit）芯片替代 FPGA 电路，同时推出基于 PXIe 总线通用标准的相应产品，达到产品系列化的目的。此外，目前应用 App 软件采用单线程方式设计，进一步将采用多线程方式设计，编制采用多线程方式对多个开关并发操作的样例程序，并开展同一交换机系统多个微波开关同时控制操作的试验，优化改进应用 App 的并发性性能。项目前期的开发工作作为后期的演进设计奠定了良好的基础。

### 参考文献：

[1] 方 芳, 吴明阁. “星链”低轨星座的主要发展动向及分析 [J]. 中国电子科学研究院学报, 2021, 16 (9): 933

- 936.

- [2] 邹明, 赵子骏, 魏凡. 新兴低轨卫星通信星座发展前景研究 [J]. 中国电子科学研究院学报, 2020, 15 (12): 1155 - 1162.
- [3] 彭中新, 祁振强, 钟圣, 等. “星链”在俄乌冲突中的运用分析与思考启示 [J]. 战术导弹技术, 2022 (6): 121 - 127.
- [4] 周松青. 俄乌冲突中美国对乌情报支持的不对称透明研究 [J]. 情报杂志, 2024, 43 (1): 28 - 34.
- [5] 彭玲, 池天河. 空天信息科技创新与产业发展布局 [J]. 人民论坛, 2023 (16): 18 - 23.
- [6] 蒋瑞红, 冯一哲, 孙耀华, 等. 面向低轨卫星网络的组网关键技术综述 [J]. 电信科学, 2023, 39 (2): 37 - 47.
- [7] 沈斐, 吕承丞, 张嘉璇, 等. 基于终端流量预测的低地球轨道卫星互联网资源分配策略 [J]. 通信学报, 2024, 45 (7): 21 - 36.
- [8] Wolfe F. U. S. Army demonstrating starlink integration for MQ-1C gray eagle [J]. Defense Daily, 2024: 6 - 7.
- [9] 孙剑锋, 牛旻. 2021年中国商业航天产业进展 [J]. 国际太空, 2022 (3): 36 - 39.
- [10] 杜文龙. “千帆星座”正在快速组网, 比“星链”更加合理, 月球可同步部署 [J/OL]. [2024 - 10 - 22]. <https://baijiahao.baidu.com/s?id=1813345110361164327&wfr=spider&for=pc>
- [11] SAC/TC 153. GB/T 40678 - 2021 PXI 总线模块通用规范 [S]. 北京: 中国标准出版社, 2021 - 10 - 11.
- [12] 袁国军, 李兴隆, 肖思敏, 等. 基于 PXI 标准的功率器件总剂量效应测试系统研制 [J]. 核电子学与探测技术, 2024 (3): 415 - 419.
- [13] Catic V D, Lukic N M, Salom I M, et al. An automated environment for hardware testing using PXI instrumentation and LabVIEW software [J]. Telfor Journal, 2017, 9 (2): 98 - 103.
- [14] 游磊, 梁颖, 王进, 等. 基于 LabWindows/CVI 的 PXI 数据采集卡软面板设计与实现 [J]. 计量与测试技术, 2020, 47 (7): 1 - 4.
- [15] 王法臻, 崔少辉, 王成. PCIe 设备驱动程序开发的关键技术 [J]. 现代电子技术, 2021, 44 (16): 65 - 69.
- [16] 李桢煜, 宋宇晨, 彭喜元, 等. 基于对比序列重构的卫星遥测数据异常检测方法 [J]. 仪器仪表学报, 2024, 45 (4): 17 - 26.
- [17] SAC/TC 153. GB/T 40676 - 2021 PXIe Express 总线模块通用规范 [S]. 北京: 中国标准出版社, 2021 - 10 - 11.
- [18] 施云峰. 基于 PCI 的多通道数据采集系统设计 [D]. 杭州: 浙江大学, 2025.
- [19] 邱瑞康. 基于 FPGA 的智能结构振动控制系统 [D]. 扬州: 扬州大学, 2025.
- [20] 丁如艺, 张激, 李嘉伟. 嵌入式系统平台抽象层设计与接口实现 [J]. 计算机系统应用, 2020, 29 (3): 108 - 113.
- (上接第 205 页)
- [6] 任朝晖, 刘玉麟, 姜泽宇, 等. 基于模糊增益滑模四旋翼无人机自适应容错控制 [J]. 东北大学学报 (自然科学版), 2024, 45 (2): 209 - 216.
- [7] 王君, 骆明敏. 多四旋翼无人机鲁棒全局快速终端滑模编队容错控制 [J]. 飞行力学, 2024, 42 (3): 68 - 75.
- [8] 刘勇求, 刘晓峰. 无人机舵面故障高阶滑模重构观测器设计及主动容错控制 [J]. 火力与指挥控制, 2024, 49 (4): 170 - 176.
- [9] 杜青峰, 荆武兴, 高长生, 等. 考虑滑块动态特性的变质心四旋翼无人机双回路控制 [J]. 北京航空航天大学学报, 2024, 50 (3): 861 - 873.
- [10] 胡竣耀, 陈巧玉, 童东兵, 等. 滑模控制的四旋翼无人机拓展状态观测器设计 [J]. 光电与控制, 2023, 30 (3): 15 - 19.
- [11] 仇翔, 郑潇, 姚奕, 等. 持续风补偿下四旋翼无人机滑模轨迹跟踪控制 [J]. 浙江工业大学学报, 2023, 51 (5): 544 - 552.
- [12] 肖仲平, 黄远鹏, 邱永峰, 等. 基于扩展扰动观测器的 PMSM 终端滑模控制 [J]. 电力电子技术, 2023, 57 (10): 35 - 38.
- [13] 胡印聪, 刘陆, 戴肖肖, 等. 基于自适应非奇异终端滑模的 Buck 变换器控制 [J]. 扬州大学学报 (自然科学版), 2024, 27 (1): 33 - 40.
- [14] 吉德志, 刘峰, 钮建波. 四旋翼巡检无人机固定时间滑模姿态控制 [J]. 机械科学与技术, 2024, 43 (9): 1640 - 1646.
- [15] 张敬渝, 栗保明. 基于磁链观测的永磁直线同步电机自适应非奇异快速终端滑模控制 [J]. 火炮发射与控制学报, 2024, 45 (5): 78 - 84.
- [16] 李争, 张梓豪, 王康涛, 等. 基于无模型的 PMLSM 改进自适应滑模自抗扰控制 [J]. 电机与控制学报, 2024, 28 (1): 142 - 151.
- [17] 李娜英, 许波. 基于 STESO 的 PMSM 非奇异快速终端滑模控制 [J]. 组合机床与自动化加工技术, 2024, (1): 92 - 96.
- [18] 李中奇, 张俊豪, 唐博伟. 高速列车精确停车的超扭曲非奇异终端滑模控制方法 [J]. 铁道学报, 2023, 45 (12): 83 - 91.
- [19] 陈彦杰, 于健业, 张振国, 等. 飞行作业机器人动态抓取的非奇异终端滑模自适应控制 [J]. 机械工程学报, 2023, 59 (23): 76 - 86.
- [20] 张猛, 赵希梅. PMLSM 伺服系统自适应非奇异快速终端滑模控制 [J]. 微特电机, 2023, 51 (10): 35 - 39.