

doi: 10.7690/bgzdh.2022.07.007

基于 PPC460 处理器的任务切换亚微秒设计

杨 渊, 邹祖伟

(中国工程物理研究院电子工程研究所, 四川 绵阳 621999)

摘要: 为提升操作系统的实时性, 对 PPC460 处理器进行硬件加速处理。采用 FPGA 对 PPC460 处理器架构进行核心代码移植, 修改原有 SRAM 空间为 DPRAM 并进行加固处理, 增加任务控制块(task control block, TCB)模块精准定位和监控任务状态及堆栈信息, 在操作系统发生任务调度时, 将原有串行压栈出栈模式修改为并行同时进出栈模式。测试结果表明: 该方法有效实现了针对 PPC460 处理器的任务切换亚微秒设计, 处理复杂应用的实时性与抗干扰能力得到极大提升。

关键词: 任务切换亚微秒; 实时操作系统; 硬件加速; PPC460 处理器

中图分类号: TP332 **文献标志码:** A

Sub Microsecond Design of Task Switching Based on PPC460 Processor

Yang Yuan, Zou Zuwei

(Institute of Electronic Engineering, China Academy of Engineering Physics, Mianyang 621999, China)

Abstract: In order to improve the real-time performance of the operating system, the PPC460 processor is accelerated by hardware. The core code of PPC460 processor architecture is transplanted by using FPGA, the original SRAM space is modified to DPRAM and reinforced, the task control block (TCB) module is added to accurately locate and monitor the task status and stack information, and the original serial push stack and pop stack mode is modified to parallel simultaneous in and out stack mode when the task scheduling occurs in the operating system. The test results show that the method effectively realizes the sub microsecond design of task switching for PPC460 processor, and greatly improves the real-time and anti-interference ability of processing complex applications.

Keywords: sub microsecond task switching; real-time operating system; hardware acceleration; PPC460 processor

0 引言

基于 PowerPC 架构的嵌入式处理器能够实现空间飞行器的控制管理和数据处理, 被广泛应用于资源受限嵌入式系统中。然而, 由于 PPC460 处理器主频限制在最大 100 MHz, 在实时操作系统应用中, 需对多达 64 个任务间进行频繁调度切换, 而每次任务切换调度均需保存当前 PPC460 各寄存器到该任务堆栈, 并从下一个高优先级任务堆栈中恢复各寄存器到 PPC460 处理器, 这些环节浪费了大量时间^[1]。经计算, 一次任务切换总耗时超过 4 μ s。

在 PPC460 处理器上搭载的实时操作系统, 其内核支持基于优先级抢占的任务调度策略, 支持中断嵌套, 具有快速响应的中断管理机制和高效的任务调度、上下文切换算法, 支持天花板优先级策略^[2]。在内核调度器中, 由调度器给任务分配处理器资源^[3]。每个任务由一组数据结构定义, 其中包括任务的优先级、状态、栈信息以及任务的执行上下文等^[4], 这些信息被存放在任务控制块(TCB)中。

系统中存在一个特殊的任务(idle task), 对应用不可见, 为所有任务中最低的优先级。

结合实时操作系统特点, 笔者采用 FPGA 对 PPC460 处理器架构进行核心代码移植, 修改原有 SRAM 空间为 DPRAM 并进行加固处理提高其抗干扰能力^[5], 同时方便后台进行任务调度时堆栈保存及恢复, 增加 64 个 TCB 模块监控, 精准定位各 TCB 任务状态及堆栈信息^[6], 在操作系统发生任务调度时, 通过硬件加速手段将原有串行压栈出栈模式修改为并行同时进出栈模式, 通过此方法能有效地将一次任务切换耗时 4 μ s 压缩至 <1 μ s, 大大提高系统实时性能, 有效提高 PPC460 处理器在复杂应用时的安全保障和性能保障。

1 PPC460 任务切换亚微秒设计

1.1 基于 FPGA 的 PPC460 架构移植

PPC460 处理器本身不具备并行处理压栈出栈设计, 且内部采用 SRAM 但端口存储器无法提供后

收稿日期: 2022-03-26; 修回日期: 2022-04-28

基金项目: 国防科工局基础科研项目

作者简介: 杨 渊(1995—), 女, 山西人, 硕士, 从事单片机与嵌入式系统中的软硬件技术、集成开发环境等技术研究。

E-mail: 492445923@qq.com.

台存储能力^[7]；因此，需要采用 FPGA 对 PPC460 处理器进行核心代码移植，并将其内部 SRAM 修改为双端口 DPRAM 并进行加固处理，方便后台硬件加速处理模块对各任务堆栈进行并行加速^[8]；同时，

需对 PPC460 处理器通用寄存器组进行修改^[9]，便于硬件加速模块同时保存现场及恢复现场。

在 FPGA 中对 PPC460 核心代码进行移植，总结其移植过程的架构如图 1 所示。

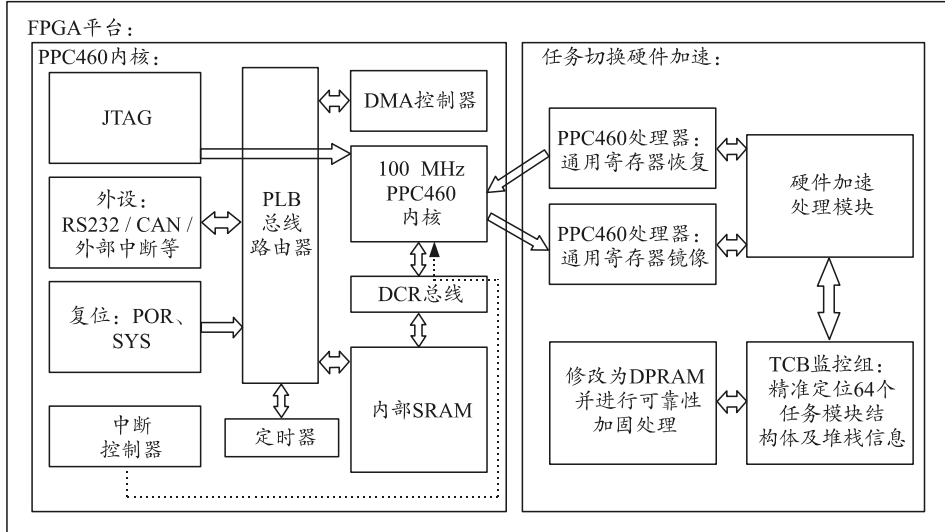


图 1 基于 FPGA 的 PPC460 架构

PPC460 包含内部 SRAM、ROM、EFLASH、DMA 控制器、PLB2OPB 等，其中内部 SRAM 模块作为整个处理器重要模块^[10]，在运行 MRCloudOS 实时操作系统所有重要数据均在 SRAM 空间内，笔者主要针对内部 SRAM 进行改进，将其修改为双端口 DPRAM 并进行加固处理，同时结合 MRCloudOS 实时操作系统特点，增加 TCB 任务监控组，精准定位 64 个任务模块各结构体及堆栈信息，通过硬件加速处理模块在发生任务切换时对 PPC460 处理器通用寄存器进行后台并发压栈出栈处理，确保任务切换时间 < 1 μs。

1.2 PPC460 增加 TCB 监控模块组

如图 2 所示，TCB 管理模块由 64 个任务监控模块组构成。

各模块工作流程为：

- 1) FPGA 逻辑实时监控任务全局优先级列表，当有任务创建或删除时，均会更新当前任务指针；
- 2) 任务创建后启动 FPGA 内并行任务监控逻辑，对 TCB 结构体内各寄存器进行监控、分析、更新就绪表等工作；基于优先级的抢占调度策略，配合 TICK 信号完成任务延时等待和任务切换调度工作；
- 3) 根据任务 TCB 结构体内堆栈寄存器实时监控各任务堆栈使用情况，对堆栈异常状态进行分析；
- 4) 各任务在监控的同时进行备份，当快照消息到来时，停止备份，当恢复消息到来时，进行场景

恢复。

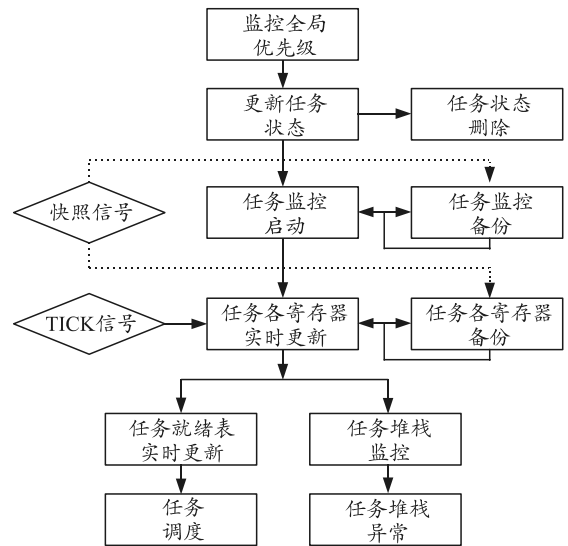


图 2 TCB 管理模块构成

1.3 PPC460 增加硬件加速处理模块

在运行实时操作系统时，为达到任务切换亚微秒设计，硬件加速处理模块需对原有串行上下文保存及恢复进行修改为并行压栈出栈方式。

如图 3 所示，实时操作系统在进行任务切换时共进行了 10 步操作：任务切换起始、更新当前任务堆栈指针、特殊寄存器压栈、32 个通用寄存器压栈、更新下一个任务堆栈指针、出栈特殊寄存器、更新 MSR 寄存器、32 个通用寄存器出栈、恢复 PC 指针、任务切换完成。

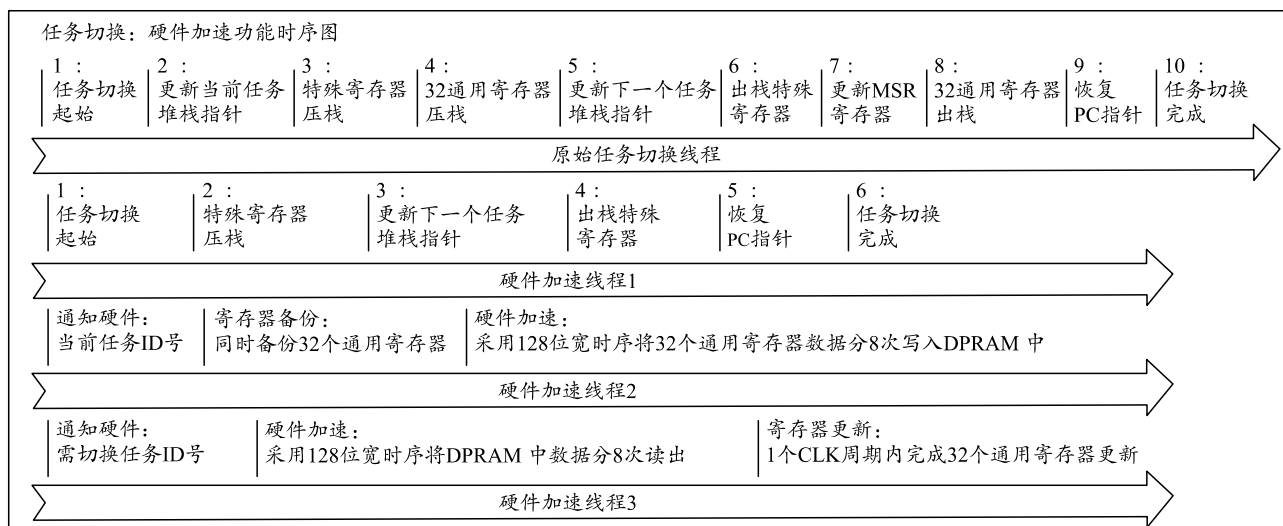


图3 硬件加速功能时序

经过分析,其主要耗时在32个通用寄存器的压栈和出栈步骤,由于实时操作系统采用32位宽的数据方式,而PPC460处理器一次数据入栈和出栈处理耗时多个CLK周期,共64次的通用寄存器入栈出栈浪费了大量的时间。

采用硬件加速处理后,将原有串行单线程方式修改为3个并行多线程方式:

线程1:任务切换起始、特殊寄存器压栈、更新下一个任务堆栈指针、出栈特殊寄存器、恢复PC指针、任务切换完成;

线程2:通知硬件当前任务ID号、硬件同时备份32个通用寄存器(只需1个CLK)、32个通用寄存器入栈(采用128位宽模式只需8个CLK);

线程3:通知硬件需切换任务ID号、硬件读出32个通用寄存(采用128位宽只需8个CLK)、更新PPC460通用寄存器(只需1个CLK)。

将原有实时操作系统任务切换时的串行流程通过硬件加速处理后变为3个并发多线程方式,通过硬件后台的128位宽改进型DPRAM解决了耗时最严重的通用寄存器压栈和出栈流程,有效提高了任务切换耗时,增加了系统的实时性。

2 实时操作系统任务切换设计

在操作系统的任务切换函数中,增加2个任务优先级的参数,这2个参数会在调用汇编时通知底层硬件代码,精准定位到要切换的2个任务的堆栈空间。按以下流程修改操作系统任务切换设计的代码。

1) 将新加入的2个参数保存到临时寄存器SPRG2/SPRG3中;

2) 发送硬核化任务切换计时命令;

- 3) 入栈特殊寄存器;
- 4) 发送硬核化压栈命令,通知硬件后台完成压栈处理;
- 5) 完成任务切换时上下文级特殊寄存器的出栈;
- 6) 发送硬核化出栈命令,通知硬件后台完成出栈处理;
- 7) 等待10个CLK(硬件1个CLK读取4个32位寄存器,32个GPR共需8个CLK,更新PPC460内核中32个处理器共需2个CLK);
- 8) 发送硬核化任务切换完成命令;
- 9) 更新PC指针。

修改后的代码将原有的压栈和出栈部分删除,增加了硬核化入栈指令和硬核化出栈指令,通知FPGA硬件加速模块进行加速线程2和加速线程3的工作,减少了PPC460处理器对任务切换时大量的入栈和出栈处理。

3 任务调度效率亚微秒级测试验证

- 1) 新建MRCloudOS实时操作系统验证工程;
- 2) 初始化中创建空闲任务Idle Task(ID: 0);
- 3) 初始化中创建最高优先级任务(ID: 7F)用于初始化硬件信息后并挂起;
- 4) 在最高有限级任务中创建测试任务1~6,对应ID为(0xA—0xF);
- 5) 测试任务1中完成串口接收功能,并调用task_delay进入任务切换阶段;
- 6) 测试任务2~6中完成串口输出功能,并调用task_delay进入任务切换阶段;
- 7) 此后MRCloudOS实时操作系统将会在Idle Task、测试任务1~6直接进行循环切换。

原始任务切换时间信息抓图与硬件加速亚微秒设计后时间信息抓图分别如图 4、5 所示。



图 4 原始任务切换时间信息



图 5 硬件加速亚微秒设计后时间信息

由以上任务切换实时时间对比抓图可以看出，

在原始系统中任务切换耗时平均在 2.5 μs 左右，而采用硬件加速亚微秒设计后，系统任务切换耗时平均在 0.96 μs 左右，任务切换效率提高 2.5 倍以上。

4 结束语

采用 FPGA 对 PPC460 处理器架构进行核心代码移植，为方便后台进行任务调度时堆栈保存及恢复，增加 64 个 TCB 模块监控，精准定位各 TCB 任务状态及堆栈信息，在操作系统发生任务调度时，通过硬件加速手段将原有串行压栈出栈模式修改为并行同时进出栈模式。结果表明：该方法能有效地将一次任务切换耗时压缩至 < 1 μs，大大提高了系统实时性能，有效提高了 PPC460 处理器在复杂应用时的安全保障和性能保障。

参考文献：

- [1] 杨珂瑶, 颜丰琳. 一种基于 PowerPC 架构性能监测方法[J]. 信息技术与信息化, 2021(8): 107-109.
- [2] 徐向权. 硬件实时操作系统中任务切换技术的研究[D]. 青岛: 青岛大学, 2018.
- [3] 程俊强, 刘铎, 陈益. 基于 PowerPC 处理器的机载多总线单板计算机设计[J]. 测控技术, 2020, 39(11): 85-90.
- [4] 范田郴. MQX Lite 实时操作系统任务调度机制的应用研究[J]. 福建电脑, 2016, 32(9): 80-81, 88.
- [5] 袁建州, 王智晶, 于丽娜. 不使用实时操作系统实现任务定时切换[J]. 汽车电器, 2013(12): 36-37, 41.
- [6] 张宇, 王省书, 胡春生. 基于 VxWorks 的多任务实时性分析[J]. 微处理机, 2013, 34(1): 53-57.
- [7] 孙利锋. 嵌入式操作系统任务切换方法对比分析[J]. 单片机与嵌入式系统应用, 2011, 11(7): 10-12, 16.
- [8] 李雪源, 丁玮, 张拓智. 基于 PowerPC 处理器的嵌入式 Linux 系统构建研究[J]. 航空计算技术, 2020, 50(5): 107-110.
- [9] 郭寒冰. 基于 PowerPC-FPGA 架构的机载雷达任务管理设计[J]. 信息技术与信息化, 2020(6): 126-128.
- [10] 吴成陆, 贺亚龙, 薛镭. 基于 PowerPC+FPGA 的数据存储模块的设计与实现[J]. 机械与电子, 2019, 37(8): 33-35, 39.