

保护控制装置虚拟运行平台设计与实现

陈宏君, 周强, 文继锋, 徐东方, 李九虎, 周磊

(南京南瑞继保电气有限公司, 江苏南京 211102)

摘要:介绍了保护控制装置的个人计算机(personal computer, PC)机虚拟运行平台设计架构和关键技术。将装置划分为管理进程、若干实时组件和非实时组件。管理进程读取配置文件, 加载各个处理器的实时、非实时组件库, 通过虚拟的CAN总线、FIFO通信完成初始化交互。管理进程启动实时线程, 按照任务周期设置关系, 串行执行实时任务; 管理进程启动若干非实时线程, 调用非实时组件周期任务, 实现和多个调试客户端的并发通信。组件之间通过虚拟数据总线、共享内存交互数据。基于平台化的开发方法, 支持装置整机功能的模拟测试, 装置配置和程序可在实际装置和虚拟装置中运行, 提高了开发效率。

关键词:保护控制; 虚拟运行; 装置仿真; 可视化配置

中图分类号: TM744

文献标志码: A

文章编号: 2096-3203(2018)01-0113-09

0 引言

保护控制装置开发正往平台化、可视化、智能化方向发展, 需要设计开发通用硬件、软件平台, 以提高应用开发效率和装置集成能力^[1-5]。文献[1]介绍了统一先进控制保护平台(unified advanced protection control, UAPC)的硬件结构, 提出了基于分布式多处理器和组件化硬件平台, 并设计了一种分布式总线的通信架构, 提高装置内数据交换效率和灵活性。文献[2]设计的平台采用PowerPC处理器作为保护和管理通信的主控芯片, 板间采用高速以太网和低电压差分信号(low-voltage differential signaling, LVDS)进行数据交换。文献[3]设计了智能变电站保护测控装置通用平台, 并基于该平台实现了二次设备装置的重构和动态配置。文献[4]提出了基于嵌入式可编程逻辑控制器(programmable logic controller, PLC)软核的配网保护硬件平台设计方案, 实现了面向配网主设备保护功能。文献[5]介绍了可视化配置软件的设计思路, 基于模块化元件进一步提升了装置可配置能力。

在装置的开发过程中, 应用程序开发调试往往依赖于装置硬件和测试设备, 串行开发的周期长, 有时不能满足快速响应市场的需求。例如: 就地化保护装置的开发, 项目立项、装置研发、入网测试、实际投运只有8个月的时间, 硬件设计、系统软件、驱动开发、应用程序和配置的移植等工作是由多个团队同时进行, 研发人员在装置尚不具备的条件下

就需进行保护功能和配置建模验证, 故有必要实现个人计算机(personal computer, PC)快速构建等同架构的虚拟装置, 除特殊硬件性能测试外, 保护功能和人机接口通信功能可以在虚拟装置上进行测试验证。文献[6]介绍了基于风和公司开发的嵌入式实时操作系统(wind river workbench's real time operating system, Vxworks)的微机保护装置仿真平台, 能完成保护功能的静态测试, 但不具备基于波形输入测试功能, 并仅限于固定硬件架构。文献[7]基于变电站配置语言(substation configuration language, SCL)模型, 实现基于IEC 61850标准的智能电子设备(intelligent electronic device, IED)仿真系统, 可模拟IED的制造报文规范(manufacturing message specification, MMS)和面向通用对象的变电站事件(generic object oriented substation events, GOOSE)通信行为及时间响应机制。文献[8]提出了一种平台化的继电保护数字仿真系统, 实际使用取得较好效果。文献[9]通过实时数字仿真仪(real time digital simulator, RTDS)闭环实时数字仿真验证了融冰控制保护策略的可行性和有效性。文献[10-12]介绍了装置实时系统部分的虚拟同步方法, 未涉及非实时通信系统的仿真实现。而一台装置的调试除应用功能的验证外, 还需进行各种版本的通信规约、人机交互显示组件的测试。随着应用的复杂度、集成度提高和芯片级系统(system on chip, SOC)多核插件的应用, 虚拟仿真平台需要适应不同的硬件架构, 可灵活加载新的服务组件。

文中基于UAPC平台的硬件、软件的架构, 通过对硬件能力进行抽象建模, 构建了虚拟数据总线, 可灵活配置不同插件的通信方式; 通过串行执行和

收稿日期: 2017-09-05; 修回日期: 2017-10-08

基金项目: 国家高技术研究发展计划(863计划)资助项目(2015AA050101)

虚拟时标机制,实现了多块多核 CPU/DSP 的实时任务虚拟运行;通过构建 Linux 的虚拟接口和组件通信机制,可将 Linux 的实时、非实时进程等重构为 Windows 下的动态链接库,实现装置的整机仿真功能,文中介绍了虚拟运行平台的架构和关键模块设计实现思路。

1 虚拟运行平台架构设计

1.1 概念定义

元件(component):是完成特定应用功能的面向对象的一组数据和对数据进行处理的过程,处理完后输出的数据供其它元件进一步使用,元件包含输入、输出、参数、逻辑节点等内容。例如过流保护功能可以通过一个过流元件的设计来完成,其输入是采样元件的采样数据,输出是过流标志,参数是过流阈值。

组件(package):是可独立发布的二进制单元。组件是 1 个黑盒子,对外的接口主要有 API 功能和端口,其中 API 是组件对外提供的功能,端口表示组件内部调用外部其它组件功能的函数^[13]。

1.2 功能设计

虚拟运行平台支持单处理器单步调试、整机仿真运行功能,支持处理器多级任务执行和虚拟数据总线交换,并满足实际装置的数据总线同步机制^[14];支持 Linux 相关进程的等价移植重构,支持虚拟液晶调试、IEC 61850 调试、IEC 103 调试及可视化调试;支持将应用程序形成 VisualStudio 工程并编译为动态链接库。仿真输入文件包括可视化配置工具 PCS-Explorer 形成的驱动包^[14](config.txt、device.cid 等装置接口文件)、应用程序库、波形文件等,输出结果包括事件记录、波形文件等。在虚拟运行平台中的配置驱动包和下载到实际装置运行的文件完全一致,而应用程序只需修改 AD 采样的通道映射代码和部分硬件接口代码,面向应用的接口函数和系统变量保持和实际装置一致。通过条件编译,可实现装置运行和仿真运行所需的程序和配置同源维护。虚拟运行平台支持如下 3 个功能:

(1) 面向应用的单处理器的单步调试功能。基于 VisualStudio 开发环境,提供单处理器的开发库,支持输入波形文件,通过采样通道重映射,逐点读取波形数据进行运算。

(2) 面向应用的整装置虚拟运行功能。支持输入波形、数据阵列或空跑运行模式,指定 PC 机装置配置文件路径、应用程序目标路径后,启动虚拟主进程(master process, Master),加载数字信号处理

器(digital signal processor, DSP)、输入输出(input output, IO)、子进程(slave process, Slave)、通信服务等组件并进行初始化,启动任务调度和通信处理线程,通过串行执行,支持实时任务虚拟微妙级的时间精度、非实时任务的 100 毫秒级调度。仿真界面可显示上电初始化的提示信息。

(3) 面向平台开发人员的公共服务单步调试功能。提供测试程序和测试驱动包,通过操作系统适配层,虚拟了同步动态存储器(synchronous dynamic random access memory, SDRAM)、共享内存、先进先出(first in first out, FIFO)通信、pthread 接口,可基于 Win7 + VisualStudio 单步调试 Master、Slave、IEC 61850、LCD、IEC 103、DSP 等组件的主体功能。

1.3 系统组成

UAPC 平台架构如图 1 所示,包括硬件平台、系统软件和可视化集成工具。所有插件按照标准化、模块化、可扩展思想设计,采用自主设计的高速总线实现装置内部高性能 CPU 插件的横向扩展。平台系统软件包括基于嵌入式操作系统开发的若干主 CPU 服务进程、DSP 和 IO 插件的通用功能模块、驱动程序,底层驱动屏蔽硬件差异,面向应用提供完善的功能和稳定的接口,支持平滑升级。各个方向的应用人员通过可视化集成工具开发和调试应用程序。主 CPU 的进程采用一主多从架构,Master 为管理主进程,负责启动运行 Slave 子进程、液晶显示进程 LCD、通信服务 IEC 61850 等多个子进程,其中 Slave 用于处理事件、录波、LED 灯刷新、统计等功能。



图 1 UAPC 平台系统框架

Fig. 1 System framework of UAPC platform

虚拟运行平台包括:可视化配置工具、仿真主界面、调试客户端、虚拟数据处理、虚拟管理进程、实时组件、非实时组件及平台接口库,其结构如图 2 所示。

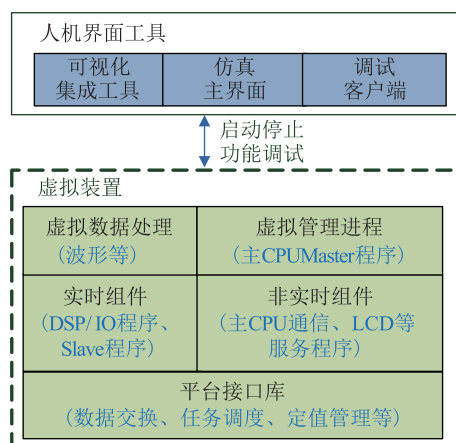


图2 虚拟运行平台结构

Fig. 2 System framework of virtual operating platform

(1) 可视化集成工具。用于嵌入式装置的硬件和软件功能配置,采用图形化符号、可视化页面搭建装置应用功能,形成装置驱动包和配置文件,和UAPC平台装置开发工具软件相同。

(2) 仿真主界面。用于仿真信息设置、启动停止仿真、虚拟装置输出信息和任务统计展示、应用程序自动编译等功能。仿真界面子系统支持设置仿真模式和仿真数据源,启动虚拟管理进程,获取虚拟管理进程的输出信息并展示。

(3) 调试客户端。包括可视化调试、虚拟液晶调试、IEC 61850 调试、IEC 103 调试等。

(4) 虚拟数据处理。为虚拟装置提供数据源,支持波形读取解析和通道映射,支持第三方仿真软件,例如直流电磁暂态计算程序/电力系统分析仿真软件(electro-magnetic transient in DC system/power system computer aided design, EMTDC/PSCAD)的输出数据,支持自定义格式的数据阵列等。

(5) 虚拟管理进程 Master。对应装置主 CPU 的管理进程 Master,读取装置配置文件,加载装置实时、非实时组件,进行实时任务、非实时任务调度,创建释放共享内存。

(6) 实时组件。对应装置各个处理器的程序功能,编译为动态链接库,此外主 CPU 的 Slave 进程也重构为实时组件。

(7) 非实时组件。对应装置主 CPU 基于操作系统开发的通信、人机交互等服务进程,包括 IEC 61850、IEC 103、LCD 等,独立编译为若干动态链接库。

(8) 平台接口库。包括 DSP/IO 实时库、主 CPU 实时库以及主 CPU 非实时库,提供装置平台化公共接口和服务,定义实时组件、非实时组件的导出

接口函数,提供数据注册、变量管理、总线通信等功能。

在图 2 中标示了虚拟运行平台和装置实际运行平台的模块映射关系:虚拟管理进程 Master 从主 CPU 的 Master 进程中移植重构,并增加了组件加载管理等内容。实时组件对应 DSP/IO/主 CPU 的 Slave 程序,根据配置的插件数量创建对应的实时组件实例。非实时组件对应主 CPU 的 IEC 61850、IEC 103、LCD 等程序,根据配置的服务进程功能列表创建对应的非实时组件实例。虚拟平台接口库包括驱动层和操作系统适配层的封装接口、公共数据管理等功能。

1.4 虚拟运行“等价”实现技术

虚拟运行平台的“等价”实现技术如图 3 所示。通过管理进程 Master 加载各类组件,完成虚拟装置的创建和初始化过程。实时组件之间通过实时多路同步高速传输串行总线协议(high performance time determinate multiplexed synchronous serial bus, HTM)/控制器局域网(controller area network, CAN)数据总线交互数据,Master 与 Slave、主 CPU 非实时组件之间通过共享内存交互数据,通过虚拟 FIFO 交互报文。Master 和 DSP/IO 实时组件之间通过 CAN 总线交互数据。图 3 中,Slave 从功能隶属关系上划分为主 CPU 的模块,从任务执行方式上划分为实时组件,在虚拟的实时任务中完成跨插件交换的数据帧缓存功能,并检测动作、自检、变位引用表中的变量翻转变化,形成事件记录和波形文件。

虚拟装置等价运行的关键步骤如下:

(1) PC 机仿真主界面进程获取仿真设置信息,启动嵌入式装置的虚拟管理进程。虚拟管理进程读取装置配置文件,获取硬件配置信息,加载各个处理器的实时组件,获取主 CPU 的进程配置信息,加载对应的非实时组件;管理进程读取配置文件中人机通信配置信息,创建配置共享内存;管理进程在加载组件时,提取并记录预定义的组件导出函数地址。

(2) 虚拟管理进程按照硬件声明顺序,通过虚拟 CAN 总线和实时组件进行初始化交互,管理进程汇总各个实时组件的数据交换信息,创建数据总线。虚拟管理进程按照配置文件中的非实时进程声明顺序,通过虚拟 FIFO 和非实时组件进行初始化交互。

(3) 虚拟管理进程启动虚拟实时任务调度线程,调用实时组件预定义的导出接口,串行执行各个等级的实时任务。虚拟管理进程启动多个非实时

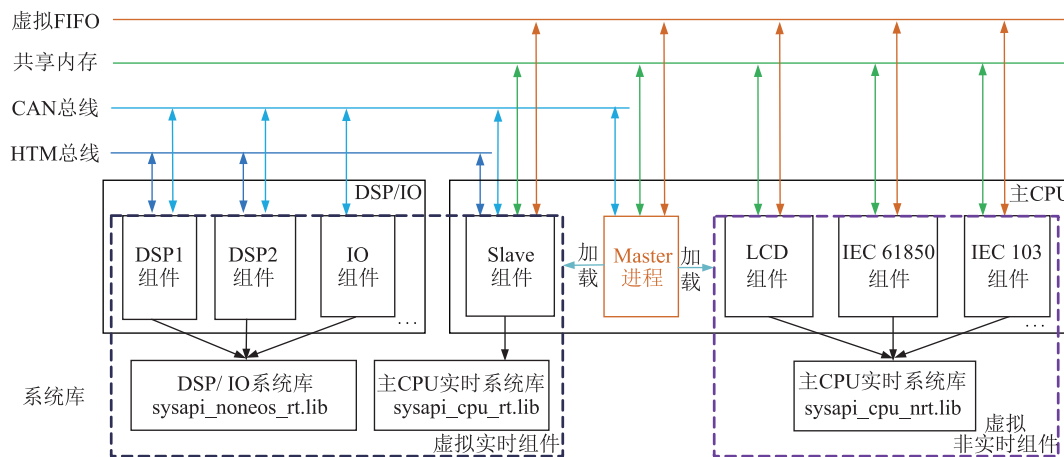


图3 虚拟运行平台等价实现示意图

Fig.3 Equivalent realization schematic diagram of virtual operating platform

线程,调用非实时组件周期任务,实现和多个调试客户端的并发通信。

(4) 虚拟装置运行过程中,实时组件基于数据总线获取处理器之间的交互数据,实时组件和非实时组件基于共享内存交互数据。实时 SLAVE 组件将非实时组件的变量代理刷新到共享内存。

2 虚拟运行平台关键模块设计

2.1 组件接口设计

实际装置可以等效为若干独立运行的进程,通过硬件中断或系统调用触发任务执行和数据同步,而虚拟装置中,如果采用若干进程模式,基于管道通信进行每个周期的同步,处理复杂度高,开销大。而若采用单个进程模式,在 1 个进程中集成多个 CPU 的程序,需要避免变量命名和函数重复,即需修改各个 CPU 的程序,增加 1 层数据封装,导致嵌入式应用程序和仿真程序有较大差异,无法实现同源维护。为降低实现复杂度,文中采用了组件模式,即将各个处理器的应用程序和 Linux 下 Slave 进程、通信进程重构改造为组件,Master 进程在初始化过程中加载组件,各个组件有独立的数据和函数空间,可解决变量空间冲突,组件之间通过预设的导出接口进行功能调用。在 Windows 下组件编译为动态链接库(dynamic load library, DLL),DLL 是一组源代码模块,每个模块包括一组可被外部模块调用的函数,模块的变量作为私有数据,只能通过接口函数访问和修改。文中通过 LoadLibrary 接口显示加载方法,在 Master 初始化过程中将 DLL 文件映射到该进程的地址空间中,操作系统为 DLL 实例分配相关变量对应的存储器,保留一段独立区域直到进程运行终止时释放。以实时组件为例,组件模型如图 4 所示。

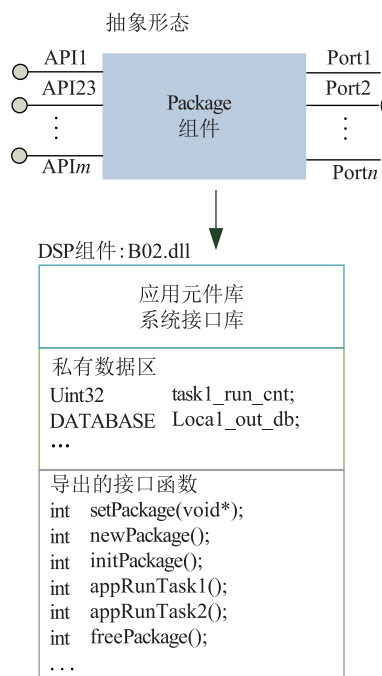


图4 组件模型

Fig.4 Modeling of package

组件的导出接口均在系统库中定义,Master 在启动过程中调用组件的导出接口,完成组件的构造过程,系统库通过预设的 appNewComponet、appInit-Component 接口,在组件的初始化过程中调用应用级元件构造、初始化接口。

2.2 平台库设计

UAPC 平台上运行的程序分为应用程序和系统程序两部分,平台系统程序为控制保护应用程序执行构建了完整的环境。在虚拟运行平台库中定义和实现组件导出接口。平台库的功能设计如下:

(1) 元件管理功能。所有应用按照元件的概念进行封装,应用功能结构清晰,具有重用性,为可视化编程配置提供了基础。基于统一的系统程序和

标准化的应用程序接口,应用元件与硬件脱离,提高了应用元件的通用性,为虚拟运行创造了必要条件。

(2) 变量注册功能。提供输入、输出变量注册接口,管理平台和应用的对外变量,支持通过层次变量名、Hash 键值快速查找获取变量属性。

(3) 参数管理功能。为了实现应用元件的可重用,同时又能够适应不同的工程的需求,通常给元件设计参数,通过改变参数的值改变元件的特性。应用元件将参数的名字、属性注册到装置参数数据库中,装置管理程序从装置配置文件中读取定值组信息,以平台结构化的参数名为索引,建立起两者之间的映射关系。

(4) 任务管理功能。平台提供任务管理和调度功能,支持应用程序创建、添加、删除、运行各个级别的任务和设置周期。应用开发人员无需关心底层任务调度,只需选择合适任务优先级即可满足各种应用的灵活需求。

(5) 平台调试功能。包括文件下载、上装、查询变量、查询内存等功能。其中查询变量的关键是获取变量的类型和地址。对于元件的注册变量,通过层次变量名在注册变量库中可获取变量信息。对于全局变量,调试客户端从编译器生成的符号表中提取。对于元件内部的静态变量,则结合元件首地址加分析元件头文件的获取的相对偏移,可得到变量信息^[14]。

虚拟运行平台库基于 UAPC 嵌入式装置系统库进行重构,其中 DSP 库可高比例复用,并统一了各种 CPU/DSP 的字长、大小端。装置为了提高运行效率,对于不同处理器采用条件编译方式静态适应部分硬件差异。在仿真模式下,构建了硬件能力描述文件,不同插件类型可灵活配置数据总线的读写能力、交换方式,根据插件型号可动态执行相关分支代码,实现仿真模式下 1 个 DSP 平台库适配多种硬件。

装置主 CPU 运行嵌入式 Linux 环境,移植到 Windows 操作系统时,关键修改有:基于开源 pthreadVC 库,原生态支持了 Linux 的线程、信号量等接口;使用共享内存模拟了 SDRAM 实时文件系统,确保了 Slave 的录波、事件等应用层代码无缝迁移;基于 windows 的确定一个或多个套接字的状态接口 (winsock API to simplify porting poll, WSAPOLL) 等应用程序编程接口 (application programming interface, API) 实现了非实时模块事件触发响应功能。移植过程中,对 gettimeofday、localtime

_r、vfscanf 等系统服务均进行了等价实现。

2.3 数据交换设计

保护控制装置的插件包括:管理 CPU 插件、计算处理插件、IO 插件。管理插件和计算处理插件之间通过高速大容量实时的 HTM 总线作为数据传输的通道^[15],这些插件与 IO 插件之间则提供了 CAN 总线作为数据传输的通道。HTM 总线上采用的是定周期传送机制,CAN 总线上采用的则是面向事件的变化传输机制。在 PC 机虚拟运行模式下,对于 CAN 总线,应用层报文保持不变,链路层通过 Master 进行中转,采用回调函数,实现接收侧直接响应。对于 HTM 总线,虚拟装置无法实现现场可编程门阵列 (field-programmable gate array, FPGA) 微秒级的快速交换周期,但需保证虚拟时间的精度以及各个组件之间的数据交换同步机理和实际装置一致。数据交换和任务调度周期存在对应关系,HTM0、HTM1 交换周期和任务 1、任务 2 周期一致,其余等级的任务周期则基于 HTM0、HTM1 扩频实现。

仿真模式下,HTM 总线基于全局共享内存实现,其原理如图 5 所示。

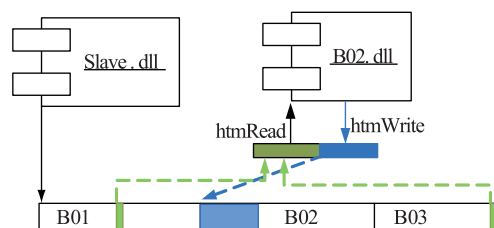


图 5 虚拟运行数据总线原理

Fig.5 Virtual operation principle of the data bus

每块插件在 HTM 周期中从共享内存读取数据,并将本插件的输出写到共享内存。实现过程如下:

(1) 每个元件在初始化阶段向系统数据库注册该应用元件可用于跨插件交换的输入、输出变量,在制作元件符号库阶段,根据输入输出接口函数,从 C 文件中提取相关信息,形成输入输出管脚。

(2) 通过可视化集成工具,从元件库中挑选元件以图形化方式搭建嵌入式装置应用程序,并通过跨页面输入、输出符号和连接线配置插件之间数据流关系,形成装置的程序驱动包。

(3) 可视化集成工具读取页面文件,根据设置的页面和元件执行顺序,依次提取插件间跨页面连接符号的连接信息(变量名、变量注册类型、变量字节、数据总线类型),形成用于插件间信号交换的输出-输入列表。

(4) 可视化集成工具离线分配变量传输地址。对各块插件以插件编号升序紧凑排序,依次处理。

对单块插件的输出变量,按照页面内元件执行顺序和变量注册类型,以自然边界对齐的原则分配变量在数据总线上的传输地址,输入变量则记录源输出变量对应的传输地址。以插件为单位,形成配置文本,记录元件实例、输出变量地址和输入变量关联地址信息。

(5) 将配置文本下载到各个插件后,由插件系统引导程序解析地址信息,在周期任务中将输出变量值刷新到对应的总线传输地址,将总线地址上数据刷新给对应的输入变量,完成变量交换。

2.4 任务执行设计

嵌入式装置基于优先级的抢占式任务调度管理,严格保证高等级任务优先执行。中断源由多个外部 IO、内部定时器和软中断组成,任务调度程序在中断驱动下,有选择地执行任务链中的任务。由于高优先级任务会打断低优先级任务执行,系统软件还提供关中断、信号量等多种手段以保证数据安全和对资源有效访问。

虚拟运行模式下,由于 PC 机定时器只能实现毫秒级的精度,并且主要目的是进行应用逻辑和装置配置的仿真,故未构建等价实时仿真系统,而是采用了串行执行和虚拟时标机制(不模拟中断嵌套),如图 6 所示,各个实时组件的相同等级任务是串行执行,以微秒为刻度更新虚拟时间,按照 HTM0 周期为步长校准虚拟微秒值、毫秒值,并将当前虚拟时间刷新到共享内存。应用程序对于展宽、延时的操作是微秒级,虚拟时标精度能够满足要求,插件间的虚拟数据交换周期和装置的实际数据总线机制是一致的,所以不需要修改数据插值同步的代码。串行执行算法在 PC 机仿真、EMTDC/PSCAD 第 3 方软件混合仿真中都进行了应用,基于波形验证仿真结果,EMTDC 一次/二次系统混合仿真结果和实际装置一致。

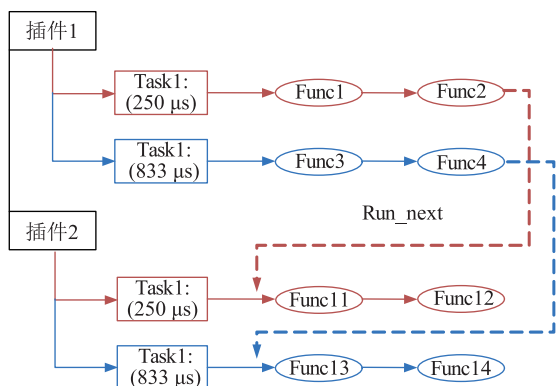


图 6 虚拟运行任务执行原理

Fig.6 Virtual operation task execution principles

2.5 实时组件设计

实时组件包括 DSP 应用组件、IO 程序组件和平台 Slave 组件。应用组件是各个 DSP 运算组件,是进行保护控制运算和逻辑处理的组件,应用组件基于实时库开发,基于元件动态注册技术^[16],通过调用应用级元件构造、初始化函数完成应用层功能实例构建。仿真设置界面支持导入嵌入式程序目录,自动形成 Makefile 和 VCProj 工程编译。装置运行程序切换到仿真模式下,需修改采样元件,调用波形通道访问接口,逐点读取数据。对于数字化 GOOSE 接收输入,则从虚拟 GOOSE 仿真输入元件中获取数据。

虚拟平台 IO 程序组件基于实时库开发,在任务队列 2 中执行,支持 CAN 数据交换功能,在可视化调试时可修改输出变量值。

虚拟平台 Slave 组件提供如下功能: FPGA 的 HTM 数据 3 s 缓存功能,通过循环读写缓冲区,缓存累计 3 s 的 HTM0、HTM1 的数据帧,用于装置启动录波时记录故障触发前数据; HTM 代理发送接收功能,代理非实时组件的跨插件交换变量的数据发送、接收,将数据刷新到共享内存; 事件报告和录波功能,在实时任务中检测信号变位,形成报文记录; 检测触发录波变量,启动录波线程,形成波形文件; LED 灯状态刷新功能、状态统计功能等。

2.6 非实时组件设计

非实时组件包括平台的公共服务,例如 LCD 液晶、IEC 61850 通信、IEC 103 通信等,在 Linux 下对应独立的进程,和 Master、Slave 通过 FIFO 交互报文。在虚拟运行模式下,上述公共服务进程改造为主 CPU 的非实时组件。将原先在 main 中主循环执行的 select-poll 任务封装为组件导出接口函数,在管理进程中启动线程执行。各非实时组件的任务线程并发执行,并根据共享内存中当前虚拟时标从 Slave 创建的 HTM 缓存区中读取数据,上送给客户端。在 Windows 下构建了基于 MMS 库的 IEC 61850 服务器端,将 MMS 库中侦听、接收的 socket 统一纳入平台的 fd 管理集,在非实时线程中处理网络通信报文,支持模型读取、数据集显示、报告上送、文件传输等服务功能^[17]。

2.7 管理进程设计

虚拟运行平台基于一主多从模式,管理进程 Master 主要功能如下:

(1) 配置解析。分 2 次解析过程,第 1 次是快读 config.txt,获取装置信息、插件配置信息、信号个数、引用表个数、定值组个数等统计内容,为分配共

享内存做准备。第2次是详细读取任务周期、信号描述、定值、数据交换配置信息、引用表等内容;

(2) 装置初始。在初始化过程中创建共享内存和数据总线,形成定值库和信号库,根据硬件配置和进程配置信息,加载相关组件,调用组件构造、初始化等接口,完成组件的数据创建和功能初始化过程。

(3) 装置运行维护。启动实时任务调度线程、非实时任务调度线程,运行过程中可视化调试通信功能;运行过程中信息显示、状态监测;运行结束后组件卸载、资源释放等。

2.8 应用实例

以保护装置 PCS-974 为例,使用 SOC 多核插件,通过平台软件 PCS-Explorer 和模块化元件库完成应用程序和 HMI 功能的配置。PCS-Explorer 形成的驱动包可下载到装置运行,也可在虚拟环境下运行,实现应用可视化程序和配置的同源开发维护。虚拟液晶调试如图 7 所示,LCD 组件增加了 TCP 通信功能,和 LCD 调试客户端通过报文交互,实现了物理液晶的显示和键盘操作响应的模拟功能。可视化调试如图 8 所示,双击连接线即可进行可视化程序页面的在线调试,提高了调试效率。

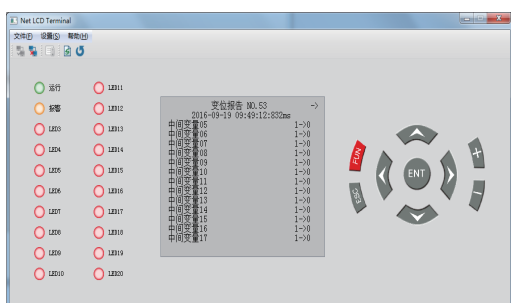


图 7 虚拟液晶调试

Fig. 7 Virtual LCD debug

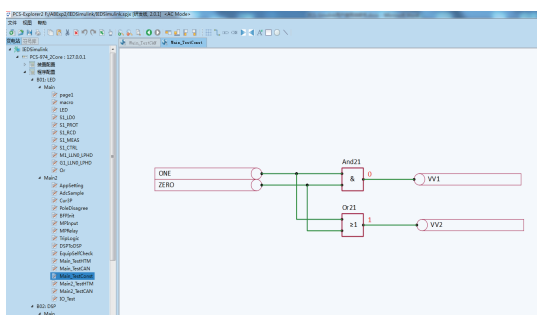


图 8 虚拟可视化调试

Fig.8 Virtual visual graph debug

3 结语

通过将虚拟装置划分为管理进程、实时模块和

非实时模块,支持装置整机功能的模拟和测试。基于平台化的开发方法,装置研发人员的配置和程序可在实际装置和虚拟装置中运行,并进行程序配置的同源维护,支持单步调试,提高了开发效率。装置研发人员在出差过程中、装置不具备的条件下,也能开发调试应用程序和验证配置,缩短了开发周期,保护控制装置虚拟运行平台在部分 PCS 装置开发中进行了应用验证。

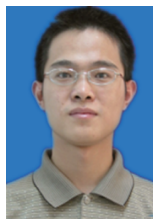
需要指出的是,虽然虚拟运行平台通过输入波形文件、EMTDC 混合仿真进行保护逻辑验证,其仿真验证结果和装置动作结果一致,但由于仿真忽略了与功能无关的硬件特性^[18],并且由于不同编译器的优化方法存在差异,不同硬件在存储结构、数据指令预取、Cache 参数也存在差异,在并发操作、时间性能评估方面,仿真运行测试无法替代实际装置运行测试。通过虚拟运行平台验证的程序和配置,归档前仍然需要进行严格的自动化测试和部分动模测试。本文下一步的研究方向是评估近似计时的仿真架构可行性,从而给出可信的程序性能统计分析信息。

参考文献:

- [1] 李响,刘国伟,冯亚东,等. 新一代控制保护系统通用硬件平台设计与应用[J]. 电力系统自动化, 2012, 36(14): 52-55.
LI Xiang, LIU Guowei, FENG Yadong, et al. Design and application of general hardware platform for new generation control and protection system [J]. Automation of Electric Power Systems, 2012, 36(14): 52-55.
- [2] 孙振华,高传发,任华锋,等. 新型基于 MPC 8309 的微机继电保护平台[J]. 计算机系统应用, 2015, 24(3): 105-109.
SUN Zhenhua, GAO Chuanfa, REN Huafeng, et al. New micro-computer relay protection platform based on MPC 8309 [J]. Computer Systems & Applications, 2015, 24(3): 105-109.
- [3] 陈翔宇,王冬青,李刚,等. 基于通用平台的智能变电站一体化设计和整合应用[J]. 电网技术, 2014, 38(增刊1): 58-62.
CHEN Xiangyu, WANG Dongqing, LI Gang, et al. Unify design and integration application device for smart substation based on general platform [J]. Power Grid Technology, 2014, 38(Supplement 1): 58-62.
- [4] 郭玮,田录林,张永良,等. 基于嵌入式 PLC 软核的通用保护平台设计与实现[J]. 电力系统保护与控制, 2014, 42(16): 122-126.
GUO Wei, TIAN Lulin, ZHANG Yongliang, et al. Design and implement of protection platform based on embedded PLC soft-core [J]. Power System Protection and Control, 2014, 42(16): 122-126.
- [5] 陈宏君,刘克金,冯亚东,等. 新一代保护测控装置配套工具软件设计与应用[J]. 电力系统自动化, 2013, 37(20): 92

- 96.
CHEN Hongjun, LIU Kejin, FENG Yadong, et al. Design and application of supporting tool software for new generation protection and measuring-control devices[J]. Automation of Electric Power Systems, 2013, 37(20):92-96.
- [6] 谢志迅, 满振海. 微机保护装置虚拟平台技术及其实现[J]. 电力系统自动化, 2011, 35(3):59-62.
XIE Zhixun, MAN Zhenhai. Technique and implementation of numerical protection relay simulator platform[J]. Automation of Electric Power Systems, 2011, 35(3):59-62.
- [7] 黄树帮, 窦仁晖, 梅德东, 等. 基于 IEC 61850 标准的通用 IED 仿真系统的设计与实现[J]. 电力系统自动化, 2012, 36(18):153-158.
HUANG Shubang, DOU Renhui, MEI Dedong, et al. Design and implementation of general IED simulation system based on IEC 61850[J]. Automation of Electric Power Systems, 2012, 36(18):153-158.
- [8] 肖碧涛, 王峰. 基于平台技术的继电保护数字仿真系统[J]. 江苏电机工程, 2015, 34(4):81-84.
XIAO Bitao, WANG Feng. Relay protection digital simulation system based on platform technology[J]. Jiangsu Electrical Engineering, 2015, 34(4):81-84.
- [9] 薛海平, 赵森林, 卢亚军, 等. 特高压直流工程的融冰控制保护策略及试验分析[J]. 电力工程技术, 2017, 36(1):84-90.
XUE Haiping, ZHAO Senlin, LU Yajun, et al. Analysis of control and protection strategy and test for de-icing mode in UHVDC project[J]. Electric Power Engineering Technology, 2017, 36(1):84-90.
- [10] 陈宏君, 刘克金, 文继锋, 等. 一种程序仿真测试方法及系统[P]. 中国:CN104881358A, 2015-09-02.
CHEN Hongjun, LIU Kejin, WEN Jifeng, et al. A program simulation test method and system [P]. China:CN104881358A, 2015-09-02.
- [11] 肖正强, 黄宁, 严岩, 等. 一种通过 PC 模拟继电保护装置多 CPU 同步运行的方式[P]. 中国:CN104063266A, 2015-09-24.
XIAO Zhengqiang, HUANG Ning, YAN Yan, et al. A method for simulating multi CPU synchronous operation of relay protection device by PC [P]. China:CN104063266A, 2015-09-24.
- [12] 陈宏君, 吴波, 刘克金, 等. 一种基于进程的仿真任务并行调度方法[P]. 中国:CN105718305A, 2016-06-29.
CHEN Hongjun, WU Bo, LIU Kejin, et al. A parallel scheduling method based on process simulation task [P]. China:CN105718305A, 2016-06-29.
- [13] 陈宏君, 刘克金, 张磊, 等. 跨平台人机交互软件组件集成框架[J]. 计算机系统应用, 2015, 24(11):52-57.
CHEN Hongjun, LIU Kejin, ZHANG Lei, et al. Cross-platform integration framework for HMI software [J]. Computer Systems Applications, 2015, 24(11):52-57.
- [14] 陈宏君, 冯亚东, 文继锋, 等. 嵌入式程序动态分配内存的调试方案设计[J]. 单片机与嵌入式系统应用, 2015, 15(9):18-21.
CHEN Hongjun, FENG Yadong, WEN Jifeng, et al. Debugging scheme of dynamic allocation memory in embedded program [J]. Microcontrollers & Embedded Systems, 2015, 15(9):18-21.
- [15] 冯亚东, 李九虎, 李彦, 等. 一种实时多路复用同步高速传输串行总线协议[P]. 中国:CN101459675B, 2012-12-12.
FENG Yadong, LI Jiuhu, LI Yan, et al. A real time multiplex high speed serial bus protocol [P]. China:CN101459675B, 2012-12-12.
- [16] 张春合, 余群兵, 陆征军, 等. 保护测控一体化装置的研制[J]. 电工技术, 2012(9):65-68.
ZHANG Chunhe, YU Qunbing, LU Zhengjun, et al. Design of protection and measuring-control device [J]. Electric Engineering, 2012(9):65-68.
- [17] IEC/TC57. Communication networks and systems for power utility automation, part 7-2: Basic information and communication structure-Abstract communication service interface (ACSI) [S]. Ed 2.0.2009.
- [18] 王盛朋, VANIA JOLOBOFF, 邓仰东. 一种基于近似计时模型的嵌入式 CPU 仿真器[J]. 计算机仿真, 2014, 31(5):238-242.
WANG Shengpeng, VANIA JOLOBOFF, DENG Yangdong. An approximate timing simulator for embedded CPUs [J]. Computer Simulation, 2014, 31(5):238-242.

作者简介:



陈宏君

陈宏君(1981—),男,硕士,高级工程师,从事保护控制平台研发工作(E-mail:chenhj@nrec.com);

周强(1979—),男,硕士,工程师,从事保护控制平台研发工作(E-mail:zhouqiang@nrec.com);

文继锋(1978—),男,硕士,研究员级高级工程师,从事继电保护和平台研发管理工作(E-mail:wenjf@nrec.com)。

Design and Implementation of Virtual Operation Platform for Protection and Control Device

CHEN Hongjun, ZHOU Qiang, WEN Jifeng, XU Dongfang, LI Jiuhu, ZHOU Lei
(NR Electric Co., Ltd., Nanjing 211102, China)

Abstract: The design architecture and key technologies of the PC virtual operation platform for a protection and control device are designed in this paper. The device is classified into management process, several real-time components and non-real-time components. The management process reads the configuration files, loads the real-time and non-real-time components libraries of each processor, and completes the initial interaction via virtual CAN bus and FIFO communication. The management process initiates the real-time thread, sets the relationship as per the task cycle, and executes the real-time tasks serially; the management process initiates several non-real-time threads, calls up non-real-time component cycle tasks, and realizes concurrent communication with several debugging clients. Data interaction between components is realized via virtual data bus and shared memory. The platform-based development method supports the simulated testing of the complete set function. Device configuration and program may be operated in the physical device and the virtual device, thus improving the development efficiency.

Key words: protection and control; virtual operation; device emulation; visualization configuration

(编辑 钱悦)

(上接第 112 页)

Regulation-compensation Principle and Simulation Analysis in 1000 kV UHV Transformer

FENG Shun, QU Xin, WANG Yi, ZHANG Junyong, FENG Yongcai

(State Grid Henan Electric Power Corporation Maintenance Company Operation, Zhengzhou 450006, China)

Abstract: 1000 kV UHV Nanyang substation is a UHV AC demonstration substation engineering project in our country. The UHV transformer is an important equipment in the substation. There are two transformers, which were produced by different companies, using fully compensated and non-fully compensated regulation-compensation principle. The winding connection method and regulation-compensation principle of the two transformers are introduced. The Simulink simulation models of the two transformers are established. The simulation data of the voltage of the transformer confirms the correctness of the simulation model. Finally, some suggestions on the construction of UHV are put forward according to the comparison results, which provide reference for the construction of UHV at later stage.

Key words: UHV; Nanyang substation; regulation-compensation transformer; simulation analysis

(编辑 方晶)