

一种基于软件通信体系结构的波形 组件动态部署方法

石贱弟, 赵小璞, 陈俊可

(中国电子科技集团公司第七研究所, 广州 510310)

摘要: 简述了软件无线电波形及波形组件, 分析了 SCA 波形软件装配描述文件 SAD, 提出了一种基于软件通信体系结构和波形组件管理器的组件动态部署方法, 并设计实现了组件管理器; 使用组件管理器为组件的创建和部署提供统一接口, 为波形组件的动态部署操作提供灵活方便的通用接口实现; 以宽带网络波形为例, 在 vxWorks 下进行了波形组件的部署试验; 结果表明, 该方法可以满足 SCA 波形组件的动态创建和部署要求, 解决了波形组件的创建和部署操作与其它操作夹杂在一起, 耦合度高的问题, 从而提高了组件的可重用性、可移植性和可维护性。

关键词: 软件通信体系结构; 波形组件; 动态部署; 软件无线电

A Waveform Component Dynamic Deployment Method Based on SCA

Shi Jiandi, Zhao Xiaopu, Chen Junke

(No. 7 Research Institute, China Electronics Technology Group Corporation, GuangZhou 510310, China)

Abstract: The Waveform and Component of SDR is briefly introduced and the SAD file is studied. In this paper, it presents a method for dynamic deployment component based on SCA and component manager and then implements the component manager. The component manager provides generic interface to create and dynamic deploy SCA component. The result of deployment WNW components on vxWorks demonstrates the method is flexible in dynamic deployment SCA components. While the reusability, portability and maintainability of SCA components are also improved as the result of decoupling component dynamic deployment from the other operation procedures.

Keywords: SCA; waveform component; dynamic deployment; software defined radio

0 引言

软件定义无线电台^[1-2] (SDR) 已成为当今及未来无线电系统研究及发展的方向, 其核心思想是在通用的硬件平台上, 提供一种具有组件可重用、可重编程等功能的软件系统, 解决传统通信体制不统一, 协同通信能力差的问题^[3]。软件通信体系结构^[4-5] (SCA) 就是满足 SDR 要求的基础上形成的一个开放的体系结构。SCA 规范只是提出了波形组件动态部署的要求, 对于如何实现波形组件在不同的处理器上动态部署没有提供具体的实现方法, 本文提出了一种基于组件管理器的组件动态部署的方法, 并设计实现了该组件管理器。使用组件管理器为组件的创建和部署提供统一接口, 为波形组件的动态部署操作提供灵活方便的通用接口实现, 可满足 SCA 组件的动态部署需求。

1 SCA 波形与波形组件

波形用于描述软件无线电台实现从用户输入到射频输出以及反之的一整套功能, 也就是从天线到用户之间的功能集合^[6]。SCA 中的波形应用由资源组件和设备组件组成^[7]。资源组件涉及波形中数据处理和控制等软件功能的抽象, 与具体的硬件设备无关; 而设备组件涉及

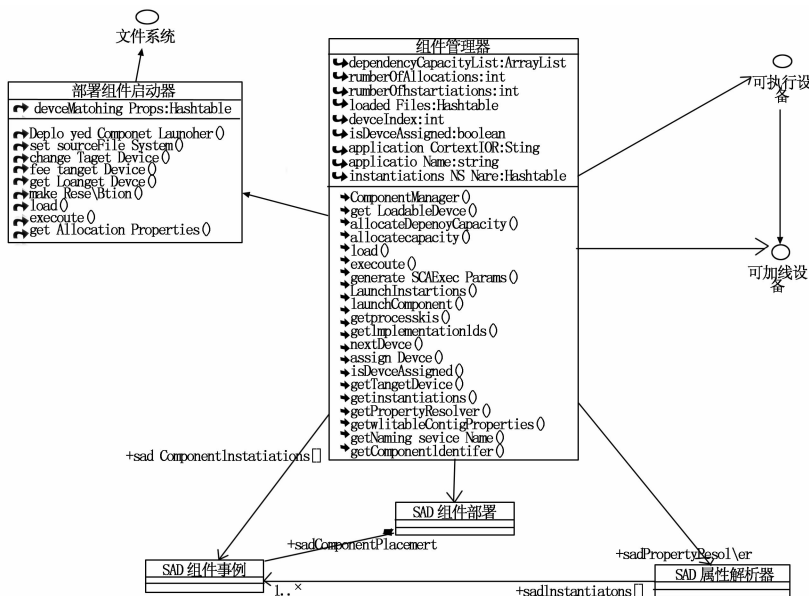


图1 组件管理器类图

波形中由专用硬件处理的部分, 它是对具体硬件设备的抽象。资源组件需要设计端口完成同其它组件的交互。资源组件除搭配控制器外通常都是进行数据处理的, 所以需要具有数据端口。数据端口分为提供端口和消费端口^[8]。SCA 规范只是规定了波形创建和部署的总体思路, 对于如何实现波形组件在不同的处理器上动态部署没有提供具体的实现技术。

收稿日期:2014-04-15; 修回日期:2014-05-18。

作者简介:石贱弟(1975-),男,广西桂林人,硕士,工程师,主要从事嵌入式系统、软件无线电技术等方向的研究。

2 基于波形组件管理器的 SCA 波形组件动态部署原理

2.1 SCA 波形软件装配描述文件

一个 SAD 实际上就对应了一个波形应用, 它描述了该波形应用需要用到的各个组件, 而每个组件的详细信息则由其对应的 SPD 描述, 组件提供或消费的接口信息由 SPD 相关联的 SCD 来描述。SAD 文件是波形创建时首先要读取的文件。SAD 包含了组成波形应用的所有组件的相关信息, 用以描述自身的装配以及组成组件间的互联。它为域的管理提供了 4 种类型的基本信息: 组件配置的需求; 软件装配时的装配控制器; 组件间的连接信息; 应用对外的可见端口。在波形工厂创建波形应用时, 需要用到 SAD 文件中的信息对应用程序中的各个组件进行初始化和配置。一个波形组件可能会包含多个资源组件, 如资源组件 R1、R2、R3、R4。每个资源组件都会对应一个软件包描述文件 SPD, 而每一个资源组件都可以有一个或多个实例程序。SAD 文件还定义了这些组件实例之间的连接关系。

2.2 组件管理器

本文设计了一个组件管理器, 用于实现对波形组件的动态部署。组件管理器类图如图 1 所示, 它继承自部署组件启动器类。

组件管理器的主要成员函数有:

- 1) 函数 `getLoadableDevice()`, 主要用于获取系统中的可加载逻辑设备以便用于加载部署波形组件。
- 2) 函数 `allocateDependencyCapacity()`, 主要用于根据组件依赖的能力进行可加载逻辑设备的选择。通过指定运行组件所需要的操作系统类型、处理器类型、中间件的类型等可以将组件部署到特定的硬件板上。
- 3) 函数 `allocateCapacity()`, 主要用于从处理器的处理资源中分配出一定的处理能力资源如内存大小、MIPS 大小给资源组件使用。如果在部署时处理器的处理资源少于资源组件需要的资源大小, 则该资源组件就不会部署到此处理器或专用器件 (如: DSP、FPGA) 上。
- 4) 函数 `load()`, 主要用于加载二进制文件到通用处理器或 DSP、FPGA 等专用处理器上。
- 5) 函数 `execute()`, 主要用于在通用处理器上运行已经加载的某个特定二进制文件。
- 6) 函数 `assignDevice()`, 用于为组件启动器分配一个可加载逻辑设备。组件启动器将会使用该分配的设备来启动波形组件。
- 7) 函数 `launchComponent()`, 用于启动组件启动器所管理的组件。在该函数内部会进行能力值的解析和使用函数 `launchInstantiations()` 函数启动该组件相关的所有实例。
- 8) 函数 `launchInstantiations()`, 用于启动组件所有的实例程序, 该函数不仅加载和执行所有的实例程序, 还会使用 `getPropertyResolver()` 函数来获取属性解析器, 然后根据所选择的组件实现来进行属性的判断和选择。
- 9) 函数 `getNamingServiceName()`, 用于根据组件的实例 ID 来获取其在命名服务器上注册的名字。在命名服务器上, 对于每个波形组件的 CORBA 引用都有一个对应的名字。
- 10) 函数 `nextDevice()`, 用于从设备列表中, 选择一个

与波形组件需求相匹配的设备。

2.3 SCA 波形的创建和动态部署

本文提出了一种基于组件管理器的波形组件动态部署实现方法, 在该方法中使用到了组件管理器、SAD 解析器、组件连接管理等, 其基本过程如图 2 所示。

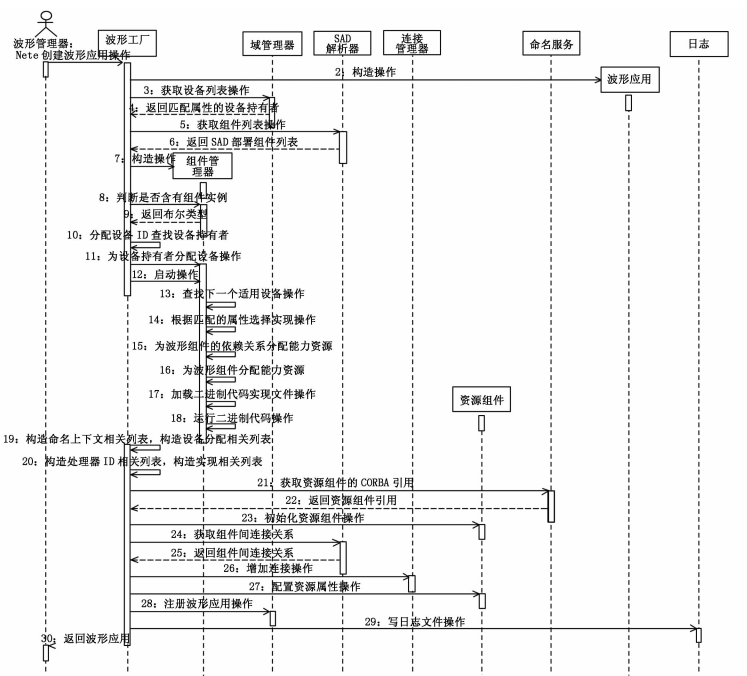


图 2 基于组件管理器的波形创建和组件动态部署序列图

波形管理器客户端程序调用了波形工厂的 `create()` 操作, 要求创建一个波形应用; 波形工厂通过调用波形应用的构造函数创建好一个波形应用; 波形工厂评估域描述文件中波形要求的内存大小、处理能力、依赖的资源、依赖的库等信息, 查找满足要求的设备; 域管理器返回匹配属性的设备持有者; 波形工厂调用 SAD 文件解析器的相关操作, 获取波形组件列表; SAD 解析器返回所有符合要求的波形组件列表; 波形工厂通过调用组件管理器的构造函数, 创建一个组件管理器; 波形工厂调用组件管理器的相关操作, 判断是否组件实例; 组件管理器返回是否含有组件实例布尔值; 波形工厂根据分配设备 ID 查找设备持有者; 波形工厂调用组件管理器相关操作, 为设备持有者分配设备; 波形工厂调用组件管理器的启动组件操作, 并启动组件所有的实例程序; 组件管理器调用相关操作, 查找下一个适用的设备; 组件管理器根据匹配的属性选择组件的实现方式; 组件管理器调用 `allocateDependencyCapacity()` 函数, 根据组件依赖的能力进行可加载逻辑设备的选择。

16 组件管理器调用 `allocateCapacity()` 函数, 分配设备的内存资源和处理能力资源给需要的波形组件; 组件管理器调用 `load()` 函数, 动态加载二进制文件到通用处理器 GPP 或专用处理器 DSP、FPGA 上; 组件管理器调用 `execute()` 函数, 在设备上运行波形组件代码, 波形组件成功部署到该设备上; 波形工厂构造命名上下文相关列表和构造设备分配相关列表; 波形工厂构造处理器 ID 相关列表和构造实现相关列表; 波形工厂通过命名服务, 获取波形组件的 CORBA 引用; 命名服务返回资源组件的 CORBA 引用; 波形工厂调用波形组件的

初始化操作, 对其进行动态部署后的自动初始化; 波形工厂通过 SAD 文件解析器, 获取波形组件间的连接关系; SAD 文件解析器返回波形组件间的连接关系; 波形工厂调用连接管理器的增加连接操作, 通知连接管理器对波形组件进行连接; 波形工厂调用波形组件的配置属性函数, 对其进行动态部署后的首次属性配置操作; 波形工厂调用域管理器的注册波形应用函数, 把刚创建的波形注册到域管理器中; 波形工厂把创建波形和部署波形组件过程中的日志信息写入日志; 波形工厂返回刚创建的波形应用给波形管理器客户端。

3 组件动态部署实例

下面以宽带网络波形为例, 说明各个组件在系统中是如何动态部署的。宽带网络波形包含如下资源组件和逻辑设备: 装配控制器组件、网络组件、编解码组件、交织解交织组件、逻辑链路控制组件、媒体访问控制组件、以太网逻辑设备、可执行逻辑设备、硬件抽象层设备等。

以太网逻辑设备是以太网设备的适配器, 供波形组件通过 CORBA 方式访问电台的以太网硬件。它运行在红边通用处理器上。可执行逻辑设备用于加载和运行波形资源组件的二进制代码。在红边通用处理器和黑边通用处理器都需要各运行一个可执行逻辑设备。硬件抽象层逻辑设备作为设备组件主要是对上层提供 SCA MHAL 标准接口, 方便波形组件的移植和重用。硬件抽象层逻辑设备运行在黑边通用处理器上, 用于完成 DSP 和 FPGA 波形组件的加载以及 GPP 中波形资源组件的数据与 DSP 和 FPGA 间进行数据的路由和转发。通过设备配置文件 DCD 的定义, 在电台启动时自动运行这些逻辑设备, 逻辑设备不需要动态部署。波形的资源组件可以根据用户的设置以及电台的实际运行情况进行动态部署。在该例子中, 假定用户预期让装配控制器组件和网络组件部署在红边, 而编解码组件、交织解交织组件、逻辑链路控制组件、媒体访问控制组件部署在黑边。组件的动态部署过程如上节所述, 参见图 2 所示。在组件的具体配置时需注意下列事项:

1) 在红边可执行设备的属性文件 PRF.xml 中定义简单类型的属性: 板卡号 BoardCardNo, 值为 1; 操作系统名 os_name, 值为 VxWorks; 处理器类型名 processor_name, 值为 PPC; MIPS 值为 1 000; Memory 值为 64 000。

2) 在黑边可执行设备的属性文件 PRF.xml 中定义简单类型的属性: 板卡号 BoardCardNo, 值为 2; 其他属性配置类似于红边处理器的属性文件。

3) 在装配控制器组件的软件包描述文件 spd.xml 中元素 implementation 内, 添加子元素 processor, 值为 ppc, 添加子元素 os, 值为 VxWorks, 添加子元素 dependency, type 属性名称为 mips_allocation, 设置 propertyref 的 value 值为 2; 添加一个子元素 dependency, type 属性名称为 memory_allocation, 设置 propertyref 的 value 值为 1 000; 添加子元素 dependency, type 属性名称为 boardCard_allocation, 设置 propertyref 的 value 值为 1。

4) 在网络组件的软件包描述文件 spd.xml 中元素 implementation 内, 添加子元素 processor, 值为 ppc, 添加子元素 os,

值为 VxWorks, 添加子元素 dependency, type 属性名称为 mips_allocation, 设置 propertyref 的 value 值为 50; 添加一个子元素 dependency, type 属性名称为 memory_allocation, 设置 propertyref 的 value 值为 2 000; 添加子元素 dependency, type 属性名称为 boardCard_allocation, 设置 propertyref 的 value 值为 1。

5) 其他波形组件如: 编解码组件、交织解交织组件、逻辑链路控制组件、媒体访问控制组件等的配置类似于网络组件。只是在属性 boardCard_allocation 的值设置为 2, 表示它们需要部署到黑板板上。其它属性值, 如: mips_allocation 和 memory_allocation 等值的大小需要根据该组件的处理复杂度和对内存需求的大小进行配置。

宽带网络波形安装、创建完成后组件在各个处理器上的部署图如图 3 所示。

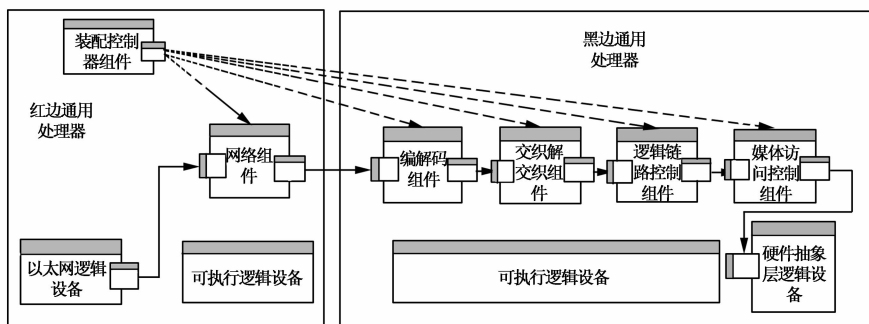


图 3 宽带网络波形组件部署图

4 结束语

本文提出了一种基于软件通信体系结构和组件管理器的 SCA 波形组件动态部署方法, 并设计实现了组件管理器。根据波形部署的需要, 需配置可执行逻辑设备属性描述文件和波形组件的软件包描述文件, 设置与波形组件动态部署相关的参数。采用组件管理器为波形组件的创建和动态部署提供了灵活方便的通用接口实现, 可以满足 SCA 波形组件的动态创建和部署的需要, 解决了波形组件的创建和部署操作与其它操作夹杂在一起, 耦合度高的问题, 从而提高了组件的可动态部署性、重用性、可移植性和可维护性。

参考文献:

- [1] Software Defined Radio Forum. Company and charter overview [EB/OL]. <http://www.sdrforum.org>, 2009.
- [2] 于全. 战术通信理论与技术 [M]. 北京: 电子工业出版社, 2009.
- [3] Place J, Kerr D, Schaefer D. Joint Tactical Radio System [A]. IEEE Military Communications Conference [C], 2000, 19: 209-213.
- [4] Joint Tactical Radio System (JTRS) Joint Program Office. Software Communication Architecture Specification [S]. SCA V2.2.2. USA: JTRS Joint Program Office, 2006, 05, 15.
- [5] 赵秋明, 林志堂, 等. 基于 SCA 的小型化核心框架研究 [J]. 计算机测量与控制, 2010, 18 (5): 1145-1147.
- [6] 杨小牛, 楼才义, 等. 软件无线电原理与应用 [M]. 北京: 北京电子工业出版社, 2001.
- [7] JTRS Joint Program Office. SCA developer's guide [EB/OL]. <http://jtrs.army.mil/sections/technicalinformation>, 2006.
- [8] Communications Research Centre Canada. SCA introduction [EB/OL]. <http://www.crc.ca/rars>, 2008.