

孟宪栋 孙富春 (清华大学计算机科学与技术系 100084)

周纬杰 (中科院软件所通用软件实验室 100080)

嵌入式 Linux 集成开发环境 研究与实现

The Research and Implementation of Integrated Development Environment for Embedded Linux

摘要: 嵌入式软件的开发是当今计算机软件发展的一个热点。本文首先分析了使用 GNU 工具链的传统嵌入式软件开发方法, 分析指出其优缺点。在此基础上介绍设计和实现能支持多种目标平台开发的嵌入式 Linux 集成开发环境的方法, 并且对核心模块和关键技术进行了介绍和分析。

关键词: 嵌入式 Linux 开发工具链 集成开发环境

1 引言

随着网络技术和数字媒体技术的发展, 嵌入式系统发展迅速。嵌入式操作系统是嵌入式系统开发的基础。近几年来, 嵌入式 Linux 因其所具有的稳定、高效、易定制、易裁减、硬件支持广泛等特点, 结合其所独具的免费、开放源代码等特征, 在嵌入式领域得到广泛应用。

在嵌入式 Linux 上进行嵌入式开发, 还必需建立完备的开发工具集, 并在此基础上实现友好、易用的开发环境, 还应该针对嵌入式平台的应用软件的开发, 为用户提供开发和调试环境, 全面提高嵌入式系统软件的开发效率和开发质量。

2 基于工具链的传统开发方法及其优缺点

GNU 编译器 (GCC) 是一个免费的 C/C++ 工具链。由于它能支持多种体系结构, 因此常被用作交叉开发工具进行嵌入式软件开发。它由一套用于编译、汇编和链接内核及应用程序的组件组成。这些组件包括: 编译器 gcc/g++ 和预处理工具 cpp、工程管理工

具 make、编译链接库 glibc、调试工具 gdb 以及一些用于操作二进制文件的实用二进制工具集等。

2.1 用 GNU 工具链进行嵌入式开发

在宿主机上编译和安装好面向目标平台的 GNU 交叉开发工具链后, 就可以利用它进行嵌入式软件开发。这通常包括嵌入式 Linux 的定制和应用软件开发两部分。

2.1.1 嵌入式 Linux 的定制和裁减

Linux 是一个通用操作系统, 包含许多某个特定嵌入式应用不需要的模块。因此可以对嵌入式 Linux 进行定制和裁减, 这样可以剔除用不到的模块, 不但能提高核心的运行效率, 而且可以减少内核所占的内存空间。这主要包括内核的定制和裁减以及基本文件系统的定制两部分工作。

2.1.2 嵌入式应用软件开发 [1, 2]

(1) 利用 vi 或者其他源代码编辑工具编辑应用程序源代码;

(2) 编写 makefile 描述多个源代码之间的依赖关系;

(3) 利用 make 工具编译和链接生成宿主平台的二进制代码;

(4) 在宿主平台仿真运行, 利用 gdb 工具进行本地调试排错;

(5) 利用 make 工具交叉编译和链接生成目标平台的二进制代码;

(6) 把二进制代码上载到目标平台, 利用 gdb 进行远程交叉调试排错;

(7) 把调试通过的应用投入实际应用, 以最终产品的方式进行验证。

一般而言, 这些开发步骤是环环相扣的, 必须在其中一个开发步骤完成后才能进行下一步的开发工作。

2.2 基于 GNU 工具链开发的优缺点

目前基于 GNU 工具链的开发方法被大多数嵌入式系统开发选用。它提供了相当完备的跨平台调试手段, 与硬件紧密结合; 而且这个免费的开发工具链性能也很好。但是这种开发方法也有一些缺点。首先, GNU 工具链都是基于控制台方式的, 在使用时要记住许多命令参数的用法, 使用很不方便; 其次, 在进行多目标平台开发时, 还必须为每个平台使用一套特定工具链; 最后, GNU 工具集成度不高, 不能把工程管理、源码编辑、编译和调试等工

作有机地结合起来, 开发效率比较低。

整个eIDE的模块结构如图1所示:

3 集成开发环境的设计与实现

随着嵌入式应用的日趋复杂, 如何提高嵌入式软件开发的效率成了人们日益关注的问题。针对基于工具链的传统开发方法的缺陷, 我们在认真研究现有多种主流嵌入式系统集成开发环境的基础上, 设计和实现了具有自主知识产权的集成开发环境eIDE。eIDE是运行于Linux上基于KDE图形库的可视化集成开发工具, 它综合了GNU工具链和Linux下的可视化开发工具的长处。eIDE与GNU工具链一样支持多目标机, 它的仿VC图形界面完全集编辑源码、项目管理、编译、下载、调试过程为一体, 整个开发过程一气呵成。与VC的调试习惯一样, 只需要鼠标几下操作, 就可以一目了然地看到函数名列表、变量、堆栈、寄存器、反汇编等调试信息。eIDE现在支持MIPS、StrongARM、x86等多种目标平台, 并且很容易扩展支持新的目标平台。

eIDE主要支持下列功能:

(1) 目标平台的配置和生成, 主要功能是嵌入式Linux内核的定制、裁减和编译生成;

嵌入式Linux基本文件系统(包括glibc、GUI运行库等)的定制和自动生成;

支持RAMDISK和JFFS2等两种文件系统映像类型。

(2) 集成的项目开发, 主要有以下功能:

应用开发向导, 帮助用户生成工程项目;

工程项目管理, 支持多工作区和多工程项目;

用户友好的编辑、编译和集成调试功能;

支持宿主环境下的编译、仿真运行和调试;

支持特定平台目标代码的生成, 上载和远程调试。

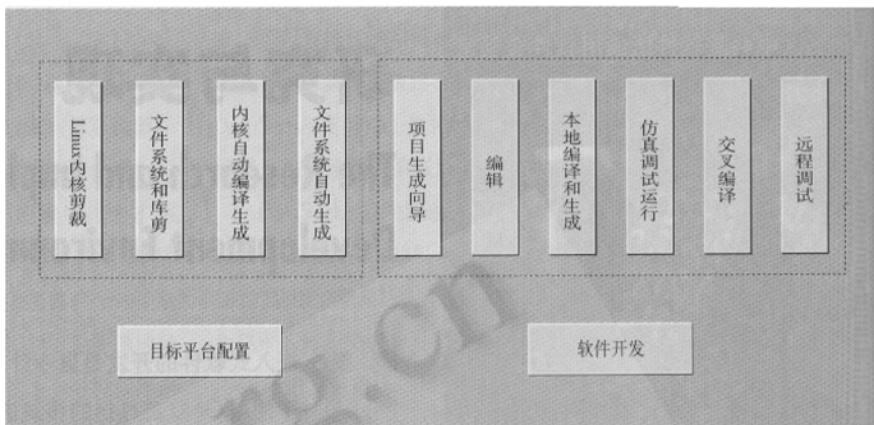


图1 集成开发环境模块

eIDE是一个高效率的、功能强大的、将多种软件开发工具紧密结合在一起的, 基于图形用户界面的嵌入式软件集成开发环境。嵌入式系统开发人员可以用它开发基于嵌入式Linux的嵌入式应用, 降低系统开发成本, 提高开发效率。下面着重阐述eIDE设计与实现中使用到的相关关键技术。

4 集成开发环境关键技术

4.1 嵌入式Linux内核的定制和裁减技术

在嵌入式系统开发中, 一般都需要针对嵌入式应用的特点对内核进行定制。但是Linux有成千上万个配置选项, 用户在配置过程中会感到无所适从, 很容易出错。考虑到这些因素, 我们采取了“视图”技术。首先对Linux配置选项进行科学分类, 确定了几大类主要的配置选项如网络、块设备支持等; 其次建立“视图”, 即这些主要配置选项和相关的标准Linux配置选项之间的单向映射关系; 最后根据建立的“视图”让用户在更高的抽象级别上进行配置。为了方便用户开发, 我们还对每个配置项附有相应的中文解释说明, 让用户用起来得心应手。

采用视图技术不但可以简化配置, 也很灵活。即使以后标准Linux配置选项发生, 也可以通过修改视图即eIDE的内核配置选项与标准内核配置选项之间的映射关系保持对用户的配置接口不变, 这也是eIDE的一个特色。

4.2 工具链集成与多平台支持技术

每个平台上的开发需要使用一套特定的工具链。但是为了给用户提供一个一致的GUI界面, 就需要把多套GNU工具链无缝集成。在eIDE中采用的方法是首先编译出每个平台特定的工具链和基本文件系统, 然后按平台分别放置到不同目录下。在全局配置文件中记录下当前支持的各个平台所在路径。在目标平台配置时根据用户选择的平台找到对应的平台根路径, 然后编译和生成对应的内核源代码和文件系统; 在应用开发时, 也是根据当前应用的目标平台选择合适的工具链和运行库进行编译、链接和调试。我们在后台开启进程调用合适的工具链并且把输入输出重定向, 从而可以把用户通过GUI设定的选项传递给工具链, 同时可以把工具链的输出有选择的在GUI上反馈。这样就可以让用户以GUI方式去使用工具链进行平台开发。

eIDE也能很方便地进行平台扩展。当需要增加平台支持时, 只要把工具链和基本文件系统

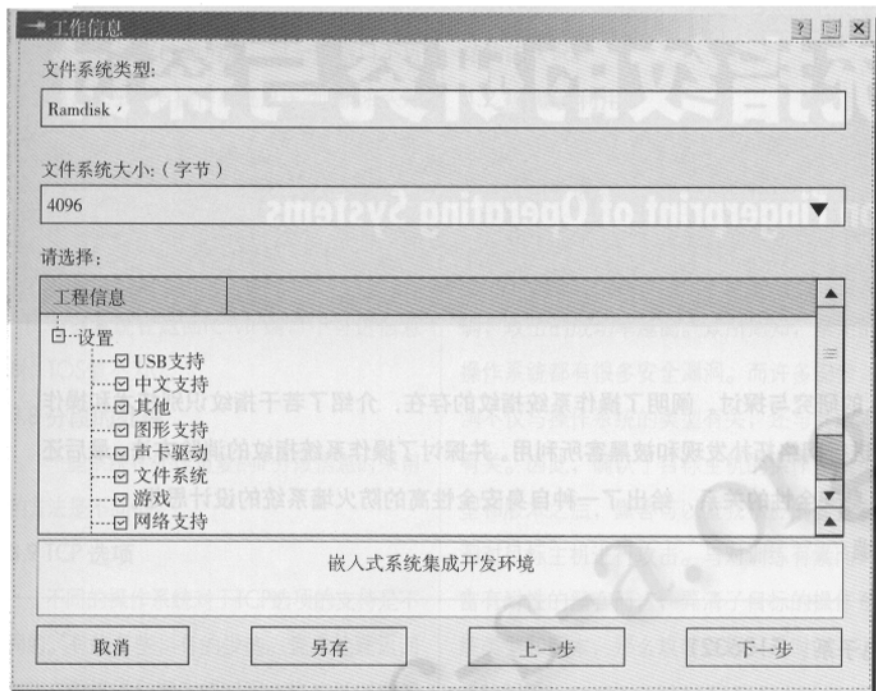


图2 文件系统配置界面

放置到一个新的平台根目录并且在配置文件中记录这些信息即可。

4.3 文件系统剪裁与生成技术

嵌入式应用的运行不仅需要嵌入式Linux内核,还需要有支持内核运行的基本文件系统和支持应用运行的相关库文件。为了减少占用不必要的存储空间,需要在基本文件系统和所有运行库的全集基础上进行剪裁。在eIDE实现中,我们把逻辑相关的多个文件当成一个逻辑单位,封装成rpm包。用户通过eIDE提供的可视化配置工具,对文件系统和库进行配置。根据用户的选择生成用户所需的文件系统,满足用户的个性化需求(如图2所示)。

4.4 远程调试技术

eIDE支持基于插桩(stub)的调试模式。它是在目标机器的操作系统和调试器内分别加入些调试功能模块,二者互通信息来进行调试。我们为支持的每个平台都实现了插桩所需的例程。如果目标操作系统支持调试的功能,可直接通过调试器进行调试功能的处理。如果目标机器不支持调试的功能,就需要由软件接管目标系统的全部异常处理(exception handler)及部分中断处理,在其中插入调试端口通信模块,与主机的调试器交互。目前支持利用串口和以太网进行调试通信。调试器需要完成的功能[3]:

- (1) 调试器与目标平台通过指定串口、网卡进行通信;
- (2) 目标操作系统对于被调试程序的所有异常处理最终都要转向通信模块,告知调试器当前的异常号;调试器据此向eIDE用户显示被调试程序产生了哪一类异常(断点、内存缺页等);
- (3) 调试器对于用户修改和控制被调试程序的请求,将转换成对被调试程序的地址空间或目标平台的某些寄存器的访问,并通过目标操作系统完成这些请求。

运行于目标平台的被调试的程序要在入口处调用一个设置断点的函数以产生异常,异常处

理程序调用调试端口通信模块,等待主机(host)上的调试器发送信息。双方建立连接后调试器便等待用户发出调试命令,目标系统等待调试器根据用户命令生成的指令。

5 总结与展望

本文介绍的嵌入式Linux的集成开发环境eIDE具有开放式、支持多种开发板、简单易用和界面友好的特点。它是针对基于GNU工具链开发方法的缺陷而开发的,它把嵌入式开发流程有机地组合起来,方便用户开发,极大地提高了开发效率。我们运用该集成开发环境成功开发了时统系统和车辆综合显控系统等多个嵌入式应用,和以往的基于工具链的开发相比开发进度和质量都得到了保证。

目前,该集成开发环境仅支持Linux宿主环境。在下一步的开发中,我们将把集成开发环境移植到Windows宿主平台下,进一步提高其易用性,并考虑支持更多的嵌入式平台,为我国嵌入式产业的发展做出新贡献。

注:本研究得到中国科学院重大支撑项目基金资助,项目编号422010303。

参考文献

- 1 邹思钦,嵌入式Linux设计与应用[M].清华大学出版社,2002。
- 2 王学龙,嵌入式Linux系统设计与应用[M].清华大学出版社,2001。
- 3 GNU国际组织,GDB用户手册[EB/OL].
<http://sources.redhat.com/gdb/>