

文章编号:1671-251X(2012)04-0104-05

AmigoBot 移动机器人上位控制系统的设计

高春生¹, 纪永刚¹, 王维², 宋锐²

(1. 铁法能源公司小青煤矿, 辽宁 调兵山 112700; 2. 山东大学机器人研究中心, 山东 济南 250061)

摘要:针对 AmigoBot 移动机器人存在不支持本地可编程控制、只能作为网络终端的问题,提出了一种基于 OMAP3530 和 Android 嵌入式平台的 AmigoBot 移动机器人上位控制系统的设计方案;给出了该系统的硬件及软件结构,详细介绍了在 OMAP3530 上移植 Android 系统的过程;首先对 Android 系统进行裁剪,使其内核功能模块既能满足上位控制系统的要求,又不冗余;在裁剪后的 Android 系统的基础上,对 AmigoBot 移动机器人的 Aria 控制软件、无线通信等应用软件进行设计。实验结果表明,该系统解决了 AmigoBot 本地控制问题,提高了 AmigoBot 的扩展能力。

关键词: AmigoBot 移动机器人; 嵌入式控制系统; 本地控制; OMAP3530; Android; Aria
中图分类号: TD67 文献标识码: A 网络出版时间: 2012-04-05 10:49
网络出版地址: <http://www.cnki.net/kcms/detail/32.1627.TP.20120405.1049.030.html>

Design of Upper Computer Control System of AmigoBot Mobile Robot

GAO Chun-sheng¹, JI Yong-gang¹, WANG Wei², SONG Rui²

(1. Xiaoqing Coal Mine of Tiefa Energy Co., Ltd., Diaobingshan 112700, China.
2. The Robotics Center of Shandong University, Jinan 250061, China)

Abstract: In view of problem that AmigoBot mobile robot did not support local programmable control and was only a network terminal, the paper proposed a design scheme of upper computer control system of AmigoBot mobile robot based on OMAP3530 and Android embedded platform. It gave hardware and software structure of the system, and introduced process of transplanting Android system on OMAP3530: firstly, clipping Android system to make its core function modules not only meet with requirements of the system, but also be irredundant; then designing application softwares of Aria control software and wireless communication of AmigoBot mobile robot on the basis of clipped Android system. The experiment result showed that the system solves problem of local control and improves expanding ability of AmigoBot.

Key words: AmigoBot mobile robot, embedded control system, local control, OMAP3530, Android, Aria

0 引言

智能机器人控制系统处理器的硬件构架和操作系统的选择对其性能优劣有着决定性的作用。AmigoBot 是一款低成本、差分驱动的面向教育的移动机器人平台^[1,2],具有 8 个声纳传感器和多种控制方式。由于 AmigoBot 移动机器人本身不能支持

本地可编程控制,只能借助网络通过 Aria 的通信协议进行通信与控制,所以为了扩展 AmigoBot 编程控制的本地化,设计 AmigoBot 移动机器人上位控制系统具有重要意义。

设计 AmigoBot 移动机器人上位控制系统时,首先,必须要考虑功耗的问题。AmigoBot 移动机器人本身由蓄电池独立供电,而且同时要向控制系

统和配备的各种传感器供应电能,所以功耗对控制平台的设计有很大的影响。其次,AmigoBot 移动机器人上位控制系统不仅单单实现 AmigoBot 的运动行为的单一任务控制,更重要的是要使其具备较复杂控制算法和控制策略的运算能力,所以对选用的处理器的运算处理能力和图像处理能力要有较高的要求。

从国内外机器人控制系统的发展来看,处理器的构架和操作系统的选择是多种多样的^[3-5]。但是处理器的构架主要以 PowerPC、ARM、DSP、FPGA 和 X86 等为主,而嵌入式操作系统则主要以 Wince、Vsworks、QNX 和 Android 等为主。PowerPC 和 ARM 一样,具有超强的运算处理能力且有较低的功耗;DSP 在图形处理方面独秀一枝;FPGA 相对来说性价比较高,但是不能完成复杂的运算任务;X86 是一种通用性的处理器,性能相对稳定,但是功耗相对 ARM 等处理器较高。综合考虑,笔者设计的 AmigoBot 移动机器人上位控制系统选用了 TI 公司的 OMAP3530 处理器。OMAP3530 是 TI 公司专为低功耗便携式应用而设计的一款嵌入式双核应用处理器^[6-7]。对操作系统而言,OMAP3530 可以支持多种操作系统,主要以 Linux 和 Wince 为主。由于 Linux 系统的体系结构比较灵活,易于裁剪,功能模块定制性高,在 Linux 新的内核版本中加入了抢占式核心,使 Linux 系统具有了软实时处理能力,且在 Linux 系统下具有较高的可靠性和安全性,而 Wince 在这些方面是无法与之相比的。因此,笔者设计的 AmigoBot 移动机器人上位控制系统采用了谷歌公司开发的 Android 嵌入式系统。该系统采用 OMAP3530+Android 的组合方式^[8],在处理器的处理能力和系统的嵌入式性能方面都达到了要求,具有较好的扩展性和实用性。

1 系统硬件设计

1.1 OMAP3530 和 Android 介绍

OMAP3530 处理器是一款高性能的双核处理器,单个芯片集成了 600 MHz ARM Cortex-A8 内核、TMS320C64x+ DSP 内核、图形引擎、视频加速器以及丰富的多媒体外设,其中,Cortex-A8 内核拥有超过当今 300 MHz ARM9 器件 4 倍的处理性能。同时 OMAP3530 是基于增强型 OMAP3 架构,可广泛应用于流媒体、2D/3D 游戏、视频会议、

高分辨率静态图像、3G 多媒体手机、高性能 PDA 等应用,它包含高性能移动产品所需的高效电源管理技术。

Android 开源系统在 Linux 核心的基础上提供了各种函数库和一个完整的应用程序框架。Android 开源系统采用了分层架构,大致分为 4 层,从低到高依次为 Linux 核心层、系统运行库层、应用程序框架层和应用程序层。Linux 核心层基于 Linux 2.6 内核,用于安全性控制、内存管理、进程管理、网络协议栈和模型驱动等。系统运行库包含了一些 C/C++ 库,被 Android 开源系统中不同的组件使用,通过 Android 应用程序框架为开发者提供服务。

1.2 硬件结构

基于 OMAP3530 和 Android 嵌入式平台的 AmigoBot 移动机器人上位控制系统(以下称上位控制系统)结构如图 1 所示。激光测距仪和 AmigoBot 本身的声纳传感器完成 AmigoBot 移动机器人的漫游障碍和地图绘制等功能;模拟摄像头用于实现 AmigoBot 运行环境的远程监控等功能。

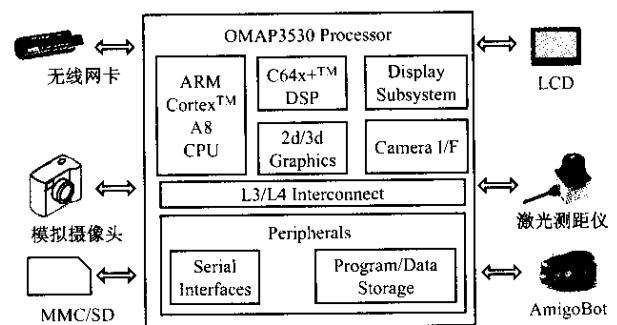


图 1 上位控制系统结构

2 Android 系统的裁剪及应用软件设计

上位控制系统是在 OMAP3530 的构架移植了 Android 系统。考虑到 Android 系统的功耗、响应效率、功能等因素,需要对 Android 系统进行必要的裁剪,使其内核功能模块既能满足上位控制系统的要求,又不冗余。在裁剪后的 Android 系统的基础上,对 AmigoBot 移动机器人的 Aria 控制软件、无线通信、激光扫描等进行开发设计。

2.1 Android 系统开发环境的构建与移植

上位控制系统的开发环境是安装在宿主机 X86 上以桌面应用为主的 Linux 操作系统 Ubuntu 9.10,其内核版本已经升级为 2.6。由于 Ubuntu 也是

一款完全遵守开源软件开发的原则,因此,对上位控制系统的开发和应用程序开发环境的搭建提供了便利。上位控制系统的软件架构如图 2 所示。

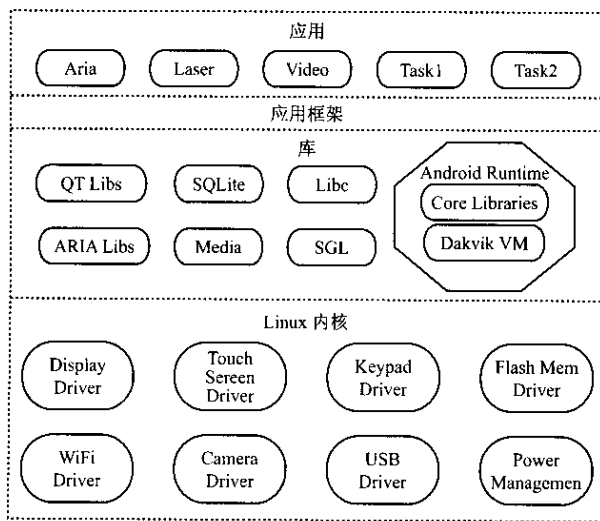


图 2 上位控制系统的软件架构

2.1.1 Ubuntu9.10 的安装与配置

在宿主机 X86 上安装了 Ubuntu9.10 的 DVD 完整版,这只是一个单纯的系统而已,想在其上面进行嵌入式的开发,必须进行必要的配置。

首先要选择一种交叉编译工具,本文使用的是 Codesourcery ARM toolchain 公司建立的 arm-none-linux-gnueabi 交叉编译工具。需要注意的是安装完成之后还是需要手动添加环境变量。Linux 是一个多用户的操作系统。多用户意味着每个用户登录系统后,都有自己专用的运行环境。而这个环境是由一组变量所定义,该组变量被称为环境变量。用户可以对自己的环境变量进行修改以达到对环境的要求。环境变量是和 shell 紧密相关的,它是通过 shell 命令来设置的。环境变量又可以被所有当前用户所运行的程序所使用。对于 bash 来说,可以通过变量名来访问相应的环境变量。添加环境变量有多种方法,既可以修改相应的环境变量的相关文件,也可以使用 shell 命令去修改。

2.1.2 Android 系统的移植

移植 Android 系统的前期工作已经准备好后,下面进行 Android 系统的移植。Android 系统的移植过程如图 3 所示。

(1) 使用 repo 命令下载对应 OMAP3 的 Android 系统的源代码

```
$ repo init -u git://labs.embinux.org/embinux-android-build/android-omap3/repo/an
$ repo sync
```

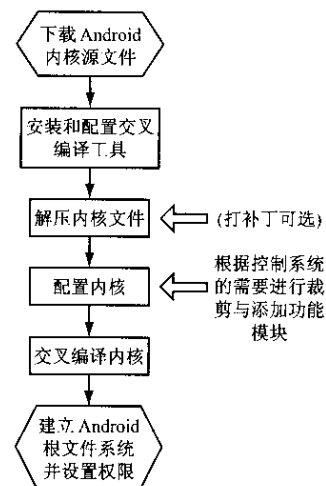


图 3 Android 系统的移植过程

(2) 配置并编译刚下载下来的内核

在进行内核编译之前需要根据上位控制系统的需要,重新配置内核,裁减掉多余的模块,添加必要的模块,如无线网卡的驱动等。

```
$ make menuconfig
$ make
```

(3) Android 系统的根文件系统 RFS

当编译完内核之后 Android 系统的 RFS 会被自动生成,但需要手动创建一个 Android 系统根目录。

```
$ cp -a root/* ~/Android_RFS
$ cp -a data/* ~/Android_RFS/data/
$ cp -a system/* ~/Android_RFS/system/
```

(4) 修改 Android 系统的 RFS 的权限

在创建完 Android 系统的 RFS 后,必须修改设置其使用权限。

```
$ cd ~/Android_RFS
$ sudo chown -R root.root *
$ sudo chmod -R 777 *
```

(5) 测试 Android 系统

在建立好 Android 系统的 RFS 后就可以从 MMC/SD 卡启动 Android 系统了,如果需要连接以太网的话,只要配置好 IP 的相关参数,就可以与网络连接。

2.2 系统驱动

根据上位控制系统的远程控制需要和图形界面编程的需要,Android 系统还需要支持 WiFi 功能和屏幕的触摸功能。

2.2.1 WiFi 功能

上位控制系统选用的是以 ZD1211b 为主芯片

的无线网卡,内核 2.6.28 版本支持该系列芯片的无线网卡,所以在配置内核时将 ZD1211b 的驱动配置到内核中,交叉编译内核后即可使用以 ZD1211b 为主芯片的无线网卡。

在 Android 系统识别到以 ZD1211b 为主芯片的无线网卡后,连接无线网络还需要必要的网络配置。Iwconfig 是 Linux Wireless Extensions (LWE)的用户层配置工具之一。LWE 是 Linux 下对无线网络配置的工具,包括内核的支持、用户层配置工具和驱动接口的支持 3 个部分。目前,很多无线网卡都支持 LWE,而且主流的 Linux 发布版本,如 Ubuntu Linux 系统本身已经内置了这个配置工具。无线网络连接程序如下:

```
$ ifconfig ra0 up //使能无线网卡,ra0 为网卡名称
$ iwlist ra0 scan //扫描附近的无线网络
$ iwpriv ra0 set SSID="Robot" //设置要连接的无线网络
$ ifconfig ra0 192.168.1.3 //设置本地无线网络连接的 IP
```

2.2.2 触摸功能

对上位控制系统来说,触摸屏的作用是非常重要的,因为一台独立的移动控制系统,不可能在其上安装键盘、鼠标等体积较大的输入设备。图形界面及触摸功能的实现流程如图 4 所示。

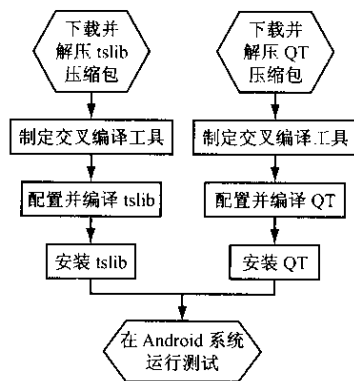


图 4 图形界面及触摸功能的实现流程

(1) 在采用触摸屏的移动终端中,由于电磁噪声的影响,触摸屏存在点击不准确、有抖动等问题,因此,在 Android 系统支持触摸屏的基础上,需要对其进行必要的滤抖、校正等操作。tslib 是一个开源的程序,能够为触摸屏驱动获得的采样提供诸如滤波、去抖、校准等功能,通常作为触摸屏驱动的适配层,为上层的应用提供了一个统一的接口。移植 tslib 的程序如下:

```
$. /autogen.sh
$. /configure --host=arm-none-linux-gnueabi
```

```
$ make&. make install
```

等待交叉编译完 tslib 后,将刚刚生成的库文件和配置文件拷贝到 MMC/SD 卡中相应的位置,最后在 Android 系统的 RFS 中添加进 tslib 的环境变量即可。

(2) 在 Ubuntu 系统下控制机器人的应用程序是采用 Qt 平台编写的,如果想要在 Android 系统上运行,必须要将 Qt 也移植进 Android 系统中。移植 Qt 的过程与移植其它程序类似,也是要经过配置、交叉编译、添加环境变量这几步:

```
$. /configure -embedded arm
```

```
$ make&. make install
```

编译完成后,将编译好的 Qt 库和 Qt 支持的字体拷贝到 MMC/FC 中 Android 系统的 RFS 的相应位置。然后根据系统的情况向 Android 系统添加环境变量。在 Android 系统上运行 Qt 测试程序,可实现图形界面触摸功能。

2.3 Aria 控制软件的移植与安装

AmigoBot 本身携带了完备的控制软件 Advanced Robotics Interface for Applications (Aria)。Aria 是一个面向对象的应用程序编程接口,它支持先锋 2/3 DX and AT、PeopleBot、PowerBot 和 AmigoBot 移动机器人。而且它采用 C++ 语言编写,通用性比较强。重要的是 Aria 是一种开源的软件,可以根据自己的需要去修改或添加其它功能。但是其只提供了 Win32 和 Linux 下的版本,所以必须进行 Aria 的跨平台移植,即将 Aria 移植到 Android 系统中。具体的移植过程如图 5 所示。

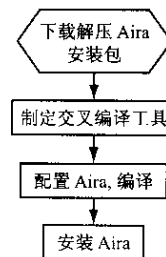


图 5 将 Aria 移植到 Android 系统中的过程

等待 Aria 交叉编译好之后,将编译后的 Aria 在 arm 平台的可执行文件和必要的库文件拷贝到 MMC/SD 卡的相应位置,添加正确环境变量和路径,在上位控制系统运行 demo 可执行文件就可以通过串口线控制 AmigoBot 移动机器人了。上位控制系统的物理链接如图 6 所示。

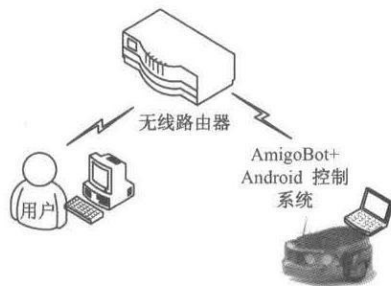


图6 AmigoBot 上位控制系统的物理链接

上位控制系统作为移动控制系统,必须具有无线控制功能,因此,下一步就要通过上位控制系统的无线功能来和远程的控制端建立连接。AmigoBot 移动机器人的远程控制采用 telnet 的方式。

telnet 协议是 TCP/IP 协议族中的一员,是 Internet 远程登录服务的标准协议和主要方式,它为用户提供了在本地计算机上完成远程主机工作的能力。在终端使用者的电脑上使用 telnet 程序,用它连接到服务器。终端使用者在 telnet 程序中输入命令,这些命令会在服务器上运行,就像直接在服务器的控制台上输入一样,从而可以在本地控制服务器。

上位控制系统采用 tp-link 无线路由搭建了一个无线的局域网,上位控制系统和远程控制计算机分别通过 ZD1211b 的无线网卡连接到无线局域网中。远程控制计算机使用 telnet 命令登录到 Android 系统,通过在 telnet 程序中输入命令来实现 AmigoBot 的远程控制。

3 结语

上位控制系统采用了 OMAP3530 双核处理器搭配 Android 嵌入式系统的组合模式。该系统搭建完成后(见图7),在实验室环境下,对其漫游模式进行了测试。测试结果表明,该系统整体功耗较低,在 AmigoBot 自带蓄电池供电的情况下,续航能力达到 3 h 以上;通过 AmigoBot 上的 8 个声纳传感器以 40 ms 的周期进行漫游蔽障的测距,满足了实时性的要求;在室内环境不间断供电的情况下,该系统可以长时间不停机工作,运行可靠。从测试结果可看

出,上位控制系统解决了 AmigoBot 本地不能在线编程控制、只能作为网络终端的缺点,扩展了其运算较复杂控制算法和控制策略的能力,而且在低功耗、可靠性等方面也满足科研需要。



图7 上位控制系统实物图

参考文献:

- [1] 刘显荣. 多机器人协作定位技术及在机器人编队中的应用[D]. 广州:华南理工大学,2011.
- [2] LIU Shuai, CHEN Chunlin, XIE Lihua. Formation Control of Multi-robot Systems [J]. Control Automation Robotics & Vision, 2010(11):1057-1062.
- [3] Heath Company. ET - 18 ROBOT Technology Manual [EB/OL]. [2011 - 12 - 17]. <http://www.robotcommunity.com/forum/thread/11565/ET-interactive-toy-or-Robot>.
- [4] DE SANTOS P G, GALVEZ J A, ESTREMER A J, et al. SIL04: A True Walking Robot for the Comparative Study of Walking Machine Techniques [J]. IEEE Robotics and Automation Magazine, 2003, 10(4):23-32.
- [5] 张生果,陈恳. 基于运动控制卡的爬壁机器人集成控制研究[J]. 电气传动, 2008,38(6):40-43.
- [6] 嵇光明,赵东标,刘凯,等. 基于 OMAP3530 的高性能数控系统设计[J]. 机械与电子,2011(12):28-31.
- [7] GOLDBERG S B. MATTHIES L. Stereo and IMU Assisted Visual Odometry on an OMAP3530 for Small Robots[J]. Computer Vision and Pattern Recognition Workshops, 2011(6):169-176.
- [8] 陈博. 基于 ARM9 S3C2440 的 Android 操作系统移植 [D]. 上海:华东师范大学,2010.