

Design of Embedded Mobile Phone Based on NiosII

CHEN Xiao-ning, ZHAO Jian, XIE Yu, QI Jin

College of Information Science and Technology, Northwest university, Xian, China

E-mail: 1.zjctec@nwu.edu.cn, 2.chxn2006@sina.com

Abstract: Nowadays embedded mobile phones are almost designed by the hard-core processor on the market. but there is the biggest question for the hard-core processor is that once the processor is made, it is hard to upgrade and enlarge its function. However, an mobile phone emulational system based on NoisII not only completes the mobile phone's basal function but also solves well the question of upgrade and enlarge function . This research successfully implemented the target that "using the software instead of the hardware design ". So ,as long as they can achieve the functions by software, developers will add or modify them at any time, but do not need to change the hardware, and it has brought great convenience for the follow-up production. In addition, this paper also gives a discussion of SOPC (System on a Programmable Chip) technology at a certain extend and proposed a relatively complete solution of SOPC.

Keywords: Soft-Core Processor; Embedded System; SOPC (System on a Programmable Chip); FPGA(Field Programmable Gate Array)

基于 NiosII 软核处理器的嵌入式移动电话设计

陈晓宁, 赵健, 谢瑜, 齐锦

西北大学 信息科学与技术学院, 西安, 中国, 710127

E-mail: 1. zjctec@nwu.edu.cn, 2. chxn2006@sina.com

【摘要】当前市面上的嵌入式移动电话主要是由硬核 (Hard core) 处理器实现的, 其存在着成品后很难进行升级和功能扩充的问题。而基于 NiosII 软核处理器设计的移动电话仿真系统不仅实现了移动电话的基本通信功能, 而且能够很好地解决产品的升级和功能扩充的问题。因此只要是能用软件实现的功能, 开发者都可以随时添加或者修改, 不必要再去硬件上的改动, 为后续的生产带来了很大的方便。此外, 本论文对 SOPC(System on a Programmable Chip)设计技术进行了一定程度的理论探讨, 并给出了一个相对完整的 SOPC 系统的实际解决方案。

【关键词】软核处理器; 嵌入式系统; 片上可编程系统; 现场可编程门阵列

1 引言

目前市面上的嵌入式移动电话主要是由硬核处理器实现的, 硬核处理器能获得广泛的应用必然有其优势。例如低成本、高性能、多样化的接口和广泛的 IP 支持厂商等。但是硬核处理器有一个最大的问题就是一旦成品后很难进行升级和功能上的扩充。而与之相比, 软核处理器具有许多其特有的优点。如其具有更大的灵活性 (Scalability) 和成本可塑性 (Cost Flexibility), 方便进行升级和功能扩充。

资助信息: 西北大学研究生重点课程《数值分析》项目基金 (09YKC21), 陕西省教育厅科技立项项目 (112Y029)

2 系统设计的核心硬件介绍

此次研究选择了 NoisII 软核处理器来实现嵌入式移动电话系统。在众多的软核处理器中之所以选择 NoisII 处理器, 是因为其有很多特有的优点。NiosII 处理器, 具有更大的灵活性和成本可塑性, 并且内嵌 NiosII 集成开发环境 (IDE), 解决了客户解决方案中成本和功能之间长期以来的矛盾, 并且更易于操作和设计。自此, 用户可以针对不同的产品定制应用不同量级的 CPU 软核, 再加上 Altera 提供的软件 SOPCBuilder 中免费附带的常用 IP 核池 (IP Cores Pool) 和 AMPP (Altera Megacore Partner Program) 等专业组织提供的商用 IP, 完全实现客户定制型产

品。鉴于以上原因，熟悉并学会使用这种新型的嵌入式系统软核设计技术是有必要和有前瞻性的。

NiosII 软核处理器是一款 32 位的 RISC(reduced instruction set computer, 精简指令集计算机)处理器，它采用 Harvard 结构（即数据存储器 and 程序存储器分开）。NiosII 处理器很独特，这首先是因为它是一种软核处理器，所以它具有一般硬核处理器没有的优势：

- 可定制特性集
- 配置系统性能
- 低成本实现

3 系统设计

3.1 系统硬件设计

3.1.1 系统硬件组成

这款基于 NiosII 软核的嵌入式移动电话主要由以下几个部分构成：

- 1) Altera 公司提供的 UP3(University Program 3) 开发板
- 2) DTP 系列 GSM 手机模块
- 3) 一个 16x2，基于字符显示的 LCD
- 4) 一个 4x10 的自制矩阵键盘
- 5) 一个用于扩展用户接口的自制扩展板

UP3 开发板是系统的核心部分。它不仅包括一块较大容量的 FPGA Cyclone EP1C6Q240C8，还包括一些嵌入式系统开发必备的存储器资源和接口。这块板子主要是用于原型设计（Prototype Design）的，并且

也支持基于 IP CORE 的设计与开发，这次设计正是利用了这一点，将 SOPCBuilder 里面的 IP 集成到板子上的 FPGA 中，实现了一个 SOPC 设计。UP3 板上已有一个嵌入式系统所需要的基本部分，为实现所需功能，需对接口 IP 进行定制。

首先需要介绍系统的硬件组成框图，如下所示：

系统硬件组成主要分为 13 个部分，其中：NiosII 软核作为该嵌入式系统的中央处理器；SRAM、SDRAM 和 FlashROM 作为系统的存储器；输入部分包括 4 个功能按钮和两个矩阵键盘接口；而输出部分则主要是 LED 和 16x2 的 LCD 模块，用于输出系统信息，可以使用户不断进行输入操作交互；通信接口主要有 3 个，一个是 JTAG UART，这个接口用于系统程序调试时在 NiosII IDE 下使用，一个是 Debug UART，这个接口用于系统与 PC 的串口调试工具之间进行调试时用，还有一个就是 GSM UART，这个接口主要控制 NiosII 对 GSM 模块指令的送出和反馈信息接收。

3.1.2 系统框架

本系统在 QuartusII 设计软件下最终以多个模块的形式呈现。主要包括 4 个部分，下面逐个作以介绍：

1) NiosII 系统模块

其输入主要包括系统时钟 (clk)、系统复位 (reset_n)，以及来自不同按钮、拨码开关、以及调试串口和 GSM 串口的输入。输出主要是前面两个串口对应的发送端口，另外主要还有两个和键盘相关的双向 PIO 端口，最后，还包括总线复用模块的输入端口（作

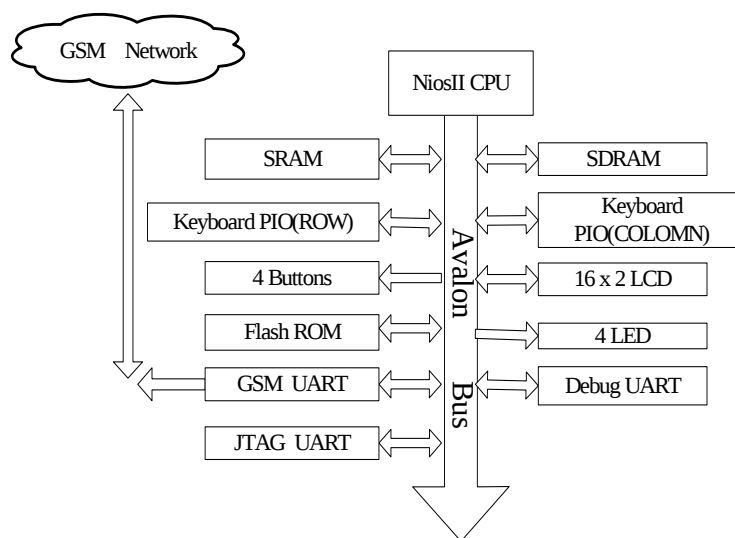


Figure 1. Systems diagrams for hardware
图 1. 系统硬件组成框图

为 NiosII 系统的输出端口)。NiosII 系统是由很多 IP 组成的, 这些 IP 都是来自 SOPCBuilder 下的 IP Pool。

2) 总线复用模块

总线复用模块是用于对 UP3 上 Cyclone FPGA 原有管脚不够的问题的一种解决方法。由于 UP3 上的 SRAM、SDRAM、FlashROM 和 LCD 有公用的数据、地址、控制等总线管脚, 因此必须将其分开来。

3) PLL 模块

PLL 模块用于对 SDRAM 芯片提供输入时钟。PLL 的原始时钟输入则来自于系统时钟 48Mhz。由于锁相环本身是用来对时钟信号进行变频的, 因此该设计直接利用了 Cyclone FPGA 的一个内嵌 PLL 模块。

4) 按钮消抖模块

此次所采用的方法是利用软件消抖, 就是使原始的输入信号经过一个多位移位寄存器, 经过几个时钟周期的延时来进行消抖。

3.1.3 GSM 与 NiosII 之间的通信

Altera 所提供的 UART IP 虽然实现了 RS-232 接口的异步时序逻辑, 但是不能直接连接或驱动一个 RS-232 串口。原因在于 Altera 大多数的 FPGA 器件不符合 RS-232 的逻辑电平规则, 如果直接连上或驱动可能会损害器件。所以连接通过 RS-232 串口通信的一个设备时, 在 Nios/NiosII 和外设接口之间要加电压转换芯片, 这样才能首先保证电气上没问题。因此扩展板上用于 GSM 通信的 RS-232 端口和 Cyclone FPGA 之间就放置了一片 MAX232。此外, 所用的调试端口是 UP3 上的一个 RS-232 接口, 它也配有电平转换芯片。

3.2 系统软件设计

3.2.1 系统设计开发环境

1) QuartusII 5.0 软件设计流程

此系统是基于 QuartusII 软件平台设计的, 其流程是这样的可以看出, 设计输入后应该对输入编译后进行确认, 而这种“确认”就是仿真原始的输入设计, 然后经过迭代的过程, 最终得到正确的结果。当然在实际生产之前还应该进行实际的板级验证, 以确保最终产品的成功。此设计就是基于这样的流程, 自顶向下 (Top-Down) 进行设计的。

2) SOPCBuilder

SOPC Builder 是一个集成在 Quartus 下的软件。

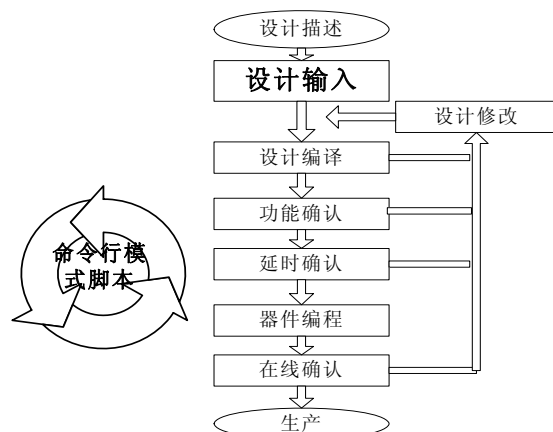


Figure 2. The design flow chart based on QuartusII software platform

图 2. 基于 QuartusII 软件平台设计流程图

更准确地说, 它是集成在 QuartusII 软件中的一个插件, 与相对独立的 NiosII IDE 不同, 它属于一种基于 IP 组装和生成系统的软件, 并没有支持开发的功能。即使使用用户自定义逻辑, 也必须把用户自己的硬件描述语言编写的文件集成进去。利用 SOPC Builder, 用户可以很方便地将处理器、存储器和其他外设模块连接起来, 形成一个完整的系统, 并且可以很容易的就生成这样一个实际的系统, 生成的系统框图最终可以在 QuartusII 的 .bdf 文件下面一模块的形式调出。

3) NiosII IDE

新一代的 NiosII 与 Nios 很重要的不同之处, 就是 NiosII 拥有自己独立的集成开发环境 NiosII IDE。NiosII IDE 本质上是开源活动 Eclipse 的一个插件, 作为 NiosII 的一部分安装。NiosII IDE 可以自主运行, 也可以以嵌在 SOPCBuilder 下面的方式打开, 所以它是一个相对独立的 IDE。

3.2.2 系统软件构成

软件构成包括 HAL 层和应用软件层两部分。

1) HAL (Hardware Abstraction Layer)

NiosII 和 Nios 最大的不同就是有了 HAL。目前 Nios II 的开发都是在 HAL 的基础上进行的。HAL 系统库是一个轻量级的运行环境, 提供了简单的硬件通讯的设备驱动程序。HAL API 集成了 ANSI C 标准库, 这些 API 允许用户用标准 C 函数去存取设备。HAL 类似于 ARM 系统中的 BSP (Board-Support Package), 提供了一个一致的设备存取界面。由于 SOPC Builder 和 NiosII IDE 紧密的集成, 在 SOPC Builder 生成硬

件系统以后, NiosII IDE 能够自动生成对应的客户 HAL 系统库。更新硬件系统设置以后, NiosII IDE 能自动更新 HAL 的驱动设置。在创建软件项目的时候, NiosII IDE 自动生成并管理。

2) 应用软件

应用软件主要指研究者编写出来的能实现将要实现功能的软件。该系统主要包括四个文件, 三个是头文件: button.h, gsm_dial_sms.h, lcd.h; 还有一个是 C 文件 system.c。下面就来分别介绍。

● gsm_dial_sms.h.h

这个头文件主要包含几个在 NiosII 与 GSM 通信时用的定义(包括宏定义和系统中准备的一些常用 AT 命令的字符形式)和一个用于向 GSM 模块发送串口命令的函数。

利用文件操作对端口进行控制的方法才是标准的 NiosII 下的端口操作方法。NiosII 下的串口编程比 Nios 下的已经简化了很多。Nios 下的串口操作要调用 Altera 自定义的函数, 而 NiosII 下, 由于有了 HAL 的存在, 串口编程已经成为标准的文件操作了。

● lcd.h

lcd.h 有两个功能, 第一, 它实现了 LCD 模块的内部寄存器访问(也就是驱动程序)和可令其他函数调用的一些功能函数; 第二, 它包含了基于 NiosII 系统开发必备的几个头文件。

● button.h

这个头文件主要包含了与键盘和按钮相关的宏定义和案件中断处理函数。

在分析 system.c 文件前有必要先说明系统的软件流程, 这样才能对最后的主文件 system.c 进行分析。系统的软件流程图如下:

● system.c

system.c 文件主要就是实现上述流程图的功能, 这个 c 文件包含前面介绍的三个头文件: lcd.h, gsm_dial_sms.h, button.h; 另外还包含一个 ANCI C 的 ctype.c 的头文件, 用它的 isdigit() 函数判断接受的输入参数是不是一个数字, 这一功能用于判断用户在应该输入电话号码时是否输入了非数字的字符, 如果是, 则系统会要求用户重新输入一个数字。

文件开头包括几个全局变量的定义:

```
alt_u8 user_number[MAX_NUMBER_LENGTH] = "";
alt_u8 message[MAX_MESSAGE_LENGTH] = "";
alt_u8 digit = 0;
```

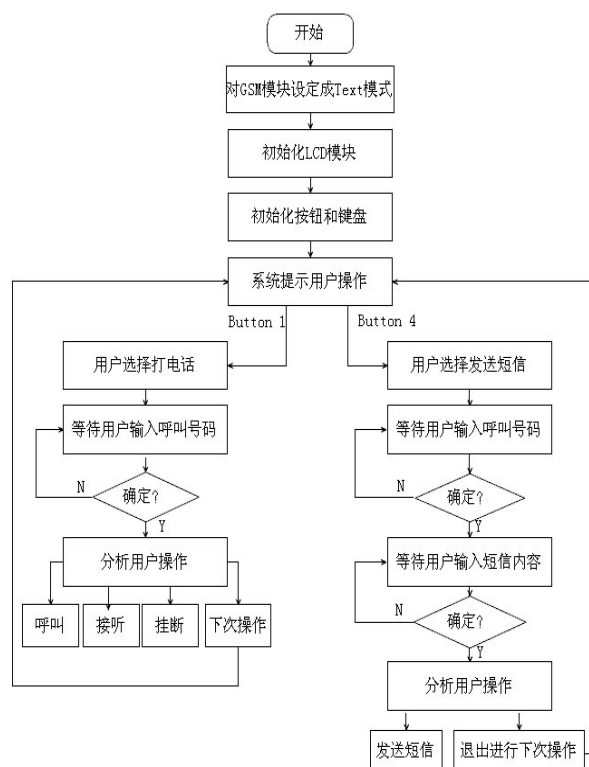


Figure3. The system software flow chart

图 3. 系统软件流程图

分别是用户输入号码的缓冲区; 用户输入短信内容的缓冲区, 以及数位 (digit), 起到数字序列中指针的功能。数组的最大限量都在前面的 gsm_dial_sms.h 头文件中有所定义。

该文件主要包含两个函数: key_scan() 以及 main() 函数, 前者用于检测键盘的按键输入, 并做出相应的操作。后者则是通过所有的头文件中的函数和宏定义来控制系统的运作流程。

4. 试验结果及分析

4.1 试验调试

此次试验采用“串口调试工具”进行调试。在进行调试操作之前已经做好了如下准备工作:

1. 连接好电源、天线和电缆。
2. 插入 SIM 卡。
3. 打开串口调试工具。
4. 设置: 波特率: 9600 bps, 数据: 8 bits, 停止位: 1, 奇偶校验: 无

上电后模块自动检测 SIM 卡和天线, 当模块找到并登录到当前可用网络, 系统就可以进行通讯。

4.2 试验分析

从调试结果来看, 此次研究, 利用软核处理器实现了 2G 移动电话的基本功能, 即双工通话、短消息的互发。成功实现了“硬件设计软件化”的目标。与目前的移动电话相比, 这种利用软核处理器代替硬核处理器的方法使系统更加容易升级和功能上的扩充。另外, 由于 NiosII 软核以及 SOPCBuilder 中所提供的各种 IP 核的可定制性, 即使是已经设计好的系统也可以在短时间内重新添加新的 IP 核和外设进行升级, 可以提高产品的市场潜在价值并减少产品的后期投入。

5. 结语

通过对系统的仿真运行以及模拟试验的结果来看, 基于 NiosII 软核处理器实现的移动电话仿真系统是可行的, 其成功实现了 2G 移动电话的基本功能, 达

到了使系统硬件设计软件化的目的。此系统的优势是为后续系统的升级及对功能的扩充带来了极大方便。

References (参考文献)

- [1] Zhou Bo, Yue Weidong, Chen Yan, Challenges of SOC SOPC Design and Practice based on NIOS [M], Tsinghua University Press, 2004
周博, 邱卫东, 陈燕, 等, 挑战 SOC 基于 NIOS 的 SOPC 设计与实践[M], 清华大学出版社, 2004
- [2] ALTERCORP. Nios Software Development Reference Manual.[EB/OL]. <http://www.altera.com.cn>, 2005
- [3] WAN Li, LI Gang, LIN Ling, A Data Acquisition System based on Nios II [J], New Technology & New Process, 2009, 03
- [4] RAO Min, QIU Dehui, LIU Yabin, The Design of a perpetual calendar Based on NiosII[J] Microcomputer Information, 2009, 14
- [5] YANG Ruirui, WANG Houjun, LI Li, Implement of Nios II Embedded System and the Development Board[J], Modern Electronics Technique, 2006, 24
- [6] DONG Liang, WANG Min, SHI Shuobiao, Realization of Chirp Function Based on NiosII Embedded System[J], Modern Electronics Technique, 2009-08