

基于 Windows CE.NET的 AD驱动程序实现与应用的研究

Implement and Application of ADC Driver Based on Windows CE.NET

舒云 邱绍峰 重庆邮电大学通信网与测试技术重点实验室 重庆 400065)

摘要

详细介绍了 S3C2410芯片 AD模块以及 Windows CE.NET的驱动模型 并且通过 AD驱动程序设计说明了流接口驱动开发方法 最后通过实例介绍了 AD驱动程序在电阻、电压等测试中的实际应用。

关键词 : S3C2410, ADC, Windows CE.NET流接口驱动程序

Abstract

In this paper, the module of ADC in S3C2410 CMOS chip and the model about Windows CE.NET drivers are expounded. The method of developing stream interface drivers are illuminated by realizing an ADC driver. In the end, application on testing resistance and voltage using ADC driver have been introduced through an example.

Keywords: S3C2410, ADC, Windows CE.NET, stream interface drivers

S3C2410芯片的 BSP中支持绝大部分的驱动程序 但有时出于实际开发的需要、操作系统的稳定性考虑 用户往往需要开发一个自己的流接口驱动程序 以下就分析了 Windows CE流接口驱动程序结构以及 AD驱动程序的开发。本驱动可以在测试电压、环阻、电流等 不涉及中断处理的驱动程序。

1 S3C2410芯片及 AD模块介绍

S3C2410是韩国三星电子公司推出的一款基于 ARM920T内核的 16/32位 RIS嵌入式微处理器 该处理器主要面向手持式设备以及高性价比、低功耗方面的应用。S3C2410芯片集成了 NAND FLASH控制器、SDRAM控制器、一个 LCD控制器支持 ST和 TFT液晶显示屏、) 个通道的 UART 个通道的 DMA 个具有 PWM功能的计时器和一个内部时钟、通道的 1位 AD同时 S3C2410还有丰富的外部接口 例如触摸屏接口、I²C总线接口、主从 USB设备接口、SPI接口、SD/MMC卡接口等等。

S3C2410内部集成了一个 通道 10位的 AD转换模块 其中第 1、第 2通道可用于支持触摸屏接口。) AD模块最大转换速率可达 500KSPS, A/D转换器以片上采样、保持的方式工作 支持掉电模式 其测量模拟输入电压范围为 -0.3~3.3V要实现 AD功能 我们需要把 AD触摸屏控制器 (ADCTSC设置成通常工作模式 (ADCTSC的 AUTO_PST=0,XY_PST=0)对 AD操作 主要对下面的 AD组寄存器进行读写操作 :

AD控制寄存器 : ADCCON(2410 ARM9台下寄存器地址是 0X58000000在 WinCE中将映射到虚拟物理地址空间);

AD触摸屏控制寄存器 : ADCTSC(2410 ARM9台下寄存器地址 0X58000004);

AD时延寄存器 : ADCDLY(2410 ARM9台下寄存器地址 0X58000008);

AD数据寄存器 : ADCDAT0/1(2410 ARM9台下寄存器地址 0X5800000C/0X58000010);

2 Windows CE.NET的驱动模型

目前基于 Windows CE.NET的驱动模型包含两种 本机设备驱动程序和流接口驱动程序。正如他们的名字所表明的 本机设备驱动程序适用于集成到基于 Windows CE.NET平台的设

备 例如 LED电源驱动就是这样的例子 他们是操作系统中 GWE\$图形窗口事件子系统 的一部分 这些驱动程序不表现在一个 DLL的动态链接库上。也有一些样本本机驱动程序 他们表现在一个 DLL的文件上 如显示、键盘驱动程序 并由 GWE\$统一管理和加载。本机设备驱动程序的结构分为两种类型 单片驱动程序和分层的驱动程序。单片驱动程序直接把硬件的功能传递给操作系统 分层驱动程序则由上层的 MDD模型设备驱动程序 和下层的 PDD依赖平台的驱动程序 组成。因为本机设备驱动程序通常与基于 Windows CE.NET的平台有着紧密的连接 微软提供了定制接口的方式来支持内部设备驱动程序 原始设备制造商 (OEM) Windows CE.NET定制到一个新平台的时候 需要创建自己的本机设备驱动程序 而对于绝大多数的开发人员来讲 则不需要编写本机设备驱动程序。本机设备驱动程序总是在 Windows CE.NET启动时加载的。

另外一种驱动模型是具有定制接口的流接口驱动程序 以 DLL动态链接库的形式存在。与本机设备驱动程序相比 所有的流接口驱动程序都是用同一组接口 并调用同一个称为流接口的函数集。当操作系统启动时 绝大多数的驱动程序 包括流接口驱动程序 是由设备管理这一进程 (DEVICE.EXE统一加载、管理和卸载的 所有这些驱动程序共享这一个进程地址空间。

每个流接口的驱动程序 都必须实现一组标准的函数 用来完成标准的文件 I/O函数和电源管理函数。

3 AD底层驱动程序的设计

AD驱动程序的主要任务 就是把 2410的 AD外设模块的使用传递给应用程序 为了便于理解 下面就按照驱动程序加载、使用的顺序 来讲述 AD底层驱动的实现。

3.1 驱动程序的加载

当操作系统启动时候 首先由设备管理器加载驱动 这时调用 ADC_Init函数。绝大多数驱动程序 都要在 XXX_Init函数中完成驱动程序的初始化 这其中包括物理地址的映射、ISP断注册、管脚的初始化等等。

在基于 Windows CE.NET的平台中 , CPU的硬件寄存器地址空间要映射到操作系统的物理寄存器地址空间 这样对操作系统物理寄存器操作如同对 CPU的硬件寄存器操作。硬件寄

寄存器地址主要由处理器决定的，但操作系统的物理寄存器地址空间是固定的，OEM在OEMAddressTable中实现了241个所有硬件寄存器到WinCE物理寄存器空间的映射。WinCE应用程序中要实现AD的物理寄存器访问，由于地址空间不同，应用程序地址空间为：0X00000000-0X41FFFFFF，操作系统物理寄存器为：0XA0000000-0XBFFFFFFF，这时就需要用到VirtualAlloc()和VirtualCopy()函数来实现地址的映射。AD初始化主要代码如下：

```
v_pADCregs=(volatile ADCReg *)VirtualAlloc (0, sizeof (ADCReg),
MEM_RESERVE, PAGE_NOACCESS)
if (v_pADCregs)
{
if! ( VirtualCopy ((PVOID)v_pADCregs, (PVOID) (ADC_BASE),sizeof
(ADCReg), PAGE_READWRITE | PAGE_NOCACHE))
{ ERRORMSG (1,(TEXT ( For ADCreg: VirtualCopy failed
\r\n )))
VirtualFree((PVOID)v_pADCregs, 0, MEM_RELEASE)
v_pADCregs = NULL
}}
```

VirtualAlloc的功能是在当前申请一段大小为 sizeof(ADCReg)的保留 (MEM_RESERVE)地址空间，VirtualCopy的功能是把一段AD物理地址，起始于ADC_BASE，大小为sizeof(ADCReg)映射到当前申请到的地址空间 (v_pADCregs)，这样驱动程序就能实现对寄存器物理地址的访问了。

3.2 打开、读、写 AD驱动程序

当应用程序打开AD驱动程序时，调用到ADC_Open()函数。这个函数比较简单，不予介绍。S3C2410有3个通道，可做一般功能使用的ADC。为了区分这3个通道，驱动程序中使用了一个全局变量SelPinNum来保存管脚号。应用程序要读取某一路的AD值时，先调用ADC_Write()函数，把这一支路号写到SelPinNum。接下来应用程序调用ADC_Read()函数来启动、读取SelPinNum这一支路的AD值。ADC_Write主要代码如下：

```
DWORD ADC_Write (DWORD hOpenContext, LPCVOID
pSourceBytes, DWORD NumberOfBytes)
{
DWORD dwRet = 2;
SelPinNum = (PWORD)pSourceBytes;
dwRet = MIN(dwRet,NumberOfBytes) // Return number of
bytes written.
return dwRet;
```

hOpenContext由CreateFile()函数返回的句柄，pSourceBytes是一个缓冲区地址，用于往驱动内写数据。这里写入的是AD支路号到SelPinNum。

应用程序接下来即可调用驱动中ADC_Read()来启动SelPinNum这一支路的AD转换，并读取转换后数据。这一过程是主要对AD物理寄存器进行操作，主要代码如下：

```
Clear_tempADCbuffer() //清空用户定义临时缓冲区
v_pADCregs->rADCDLY = ADC_DELAY_TIME //设置 AD转换
延时寄存器
for(i=0; i<MAX_SAMPLE; i++) //进行 1次 AD转换，取平均
值后保存结果
{
v_pADCregs->rADCTSC&=( 7) //设置 AD控制寄存器为正常
ADC工作模式
//下面设置 AD控制器的工作频率、如何开始转换、转换哪一支路信
号等
```

```
v_pADCregs->rADCCON&=( 7)
```

```
v_pADCregs->rADCCON=(1<<14)|((ADCPRS<<6)|((SelPin-
Num&0x7)<<3)
v_pADCregs->rADCCON|=0x1 //启动 AD转换
while(v_pADCregs->rADCCON & 0x1) //检测转换是否进行
while( (0x8000&v_pADCregs->rADCCON)) //检测
转换是否结束
ADC_Buffer[i] = 0x3ff & v_pADCregs->rADCDATA //读
取转换后数值
Sleep(Delays) //延迟一段时间以进行第二次转换
}
* (PWORD)pBuffer = SUM_AdcBuffer()/MAX_SAMPLE //返
回 1次转换后的平均值
pBuffer是驱动程序 ADC_Read()中返回给应用程序的
AD数据。
```

主要的流接口函数介绍完毕，接下来只要完成ADC_Close、ADC_Deinit等其他流接口就可以了。

3.3 流接口驱动程序的加载

有了这些编译出来的DL函数，还不能叫做流接口驱动程序，因为其流接口函数还没有导出，还需要告诉连接程序需要输出一些什么样的函数。为此就必须建立一个自己的d文件。下面是此文件的内容：

```
LIBRARY adc2410 //动态链接库名称
EXPORTS //列出将要导出函数名
ADC_Init
ADC_Deinit
ADC_Open
ADC_Close
ADC_PowerDown
ADC_IOControl
ADC_PowerUp
ADC_Read
ADC_Seek
ADC_Write
```

接下来在PB打开platform.reg设置驱动程序的加载信息：
[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\ADC2410]
Prefix = ADC //驱动设备文件名
Dll = adc2410.Dll //驱动程序的 DL动态链接库名
Order = dword:3 //驱动程序加载的时序
Index = dword:1 //驱动程序的索引号

同样在PB打开platform.bi设置把驱动加载到系统的NK.BI中，
adc2410.dll \$(_FLATRELEASEDIR)adc2410.dll NK SH
最后再增加两个文件，makefile定义了编译规则和source使用的头文件、自定义函数库、系统函数库等。重新编译Windows CE，AD驱动程序即可加载进操作系统。使用KIT把编译好的Windows CE镜像NK.BI下载到板子上使用PB的模块查看器可以看到驱动程序adc2410.dll已经被成功加载，如图1所示。

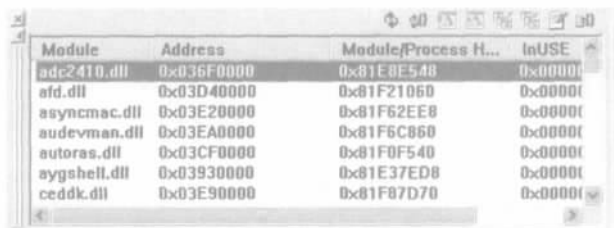


图1 PB模块查看器

驱动方式来刷新 Web 组件中来自一个或多个 InfoPlus.2 数据源的数据。Web 组件与 InfoPlus.2 数据源通讯 请求历史数据 或者应用远程数据服务 (RDS) 记录数据 与运行在 Internet 信息服务器上的过程数据 Web 目标通讯 或者与 Active 数据目标通讯。

Event.2 是 InfoPlus.2 的报警和事件跟踪系统。通过预组态的方式对报警和异常事件进行管理。用户可以用浏览器的方式进行浏览和监控。

2.4. 实时数据库接口软件

CIM- I 是一标准的接口平台 其允许实时数据库和应用软件从 DC 或 PLC 以及其它类型的实时数据库系统中读写实时数据和历史数据。

CIM- IO^M for OPC 这是实时数据库和满足服从 OPC 协议标准的控制设备的一个接口软件。它可以连接各厂的可提供 OPC 接口的设备。

CIM- IO for InTouch 这是实时数据库系统和 InTouch 系统的接口软件。

CIM- IO for Intellution 这是实时数据库系统和 InTouch 系统的接口。如动力厂的液化装置采用 Intellution 可通过 CIM- IO for Intellution 软件进行现场实时数据采集。

2.4. 客户端软件

Aspen Process Explorer 以流程图及趋势图的方式组织及显示来自 InfoPlus.2 和关系数据库的信息。这是 AspenTech 的最新的桌面 GUI 客户端产品。它是一个浏览大量实时数据的工具 一个以趋势显示的方式分析实时、历史数据的工具。同时

也是一个可以组态流程图 包括立体画面、嵌入位图、趋势图以及组织桌面应用的工具。它可以使公司生产管理者应用在线的数据实时做出有关生产、管理方面的决策。它还可以增加有关技术人员的过程知识 从而使他们对过程的变化及优化生产过程做出快速的响应。用户可以很容易通过流程图的组态来完成手动输入数据而不需要专门的软件。

Desktop ODB 这个软件用于与实时数据库交换信息。通过桌面 ODB 模块使用户应用如 Microsoft Excel Microsoft Access Crystal Report Microsoft Visual Basic 等查询实时数据库的数据 并生成统计报表。

3 结束语

整个方案的核心是 ASPEN InfoPlus.2 实时数据库。它采集、管理来自公司各生产装置控制系统的过程数据 并将这些数据整合后提供给其他应用系统 同时与以关系数据库为核心的 ER 系统集成 为 ER 系统提供所需的实时过程数据 完成了企业信息化实时数据平台的建设 成为仪化公司生产实时监控、调度的利器 系统自运行以来取得了良好的效果。

参考文献

- [1] Peter Rob, Elie Sema 等 数据库设计与开发教程 [M] 书 举 许向众 韩德强 等译 北京 电子工业出版社 ,2002
- [2] 周立柱 冯建华 等 .SQL SERVER 数据库原理—设计与实现 [M] 北京 清华大学出版社 ,2004
- [3] 焦焰 王志英 韩伟红 等 分布式数据库技术 [M] 北京 国防工业出版社 ,2000

收稿日期 :2006.11.3

上接第 5 页)

4 AD 驱动程序的使用

在基于 241 的 Windows CE 环境下 用这个驱动程序可以实现外部模拟信号到 241 数字信号的转换 下面是一个典型电压测量的运用 原理图如图 2。

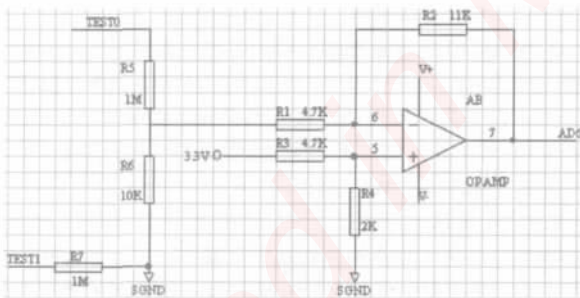


图 2 电压测量电路

TEST0/TEST1 是模拟待测量电压 经 1 倍分压后接到一个减法放大器 然后与 241 的第 6 号 AD 管脚连接 放大器可起到信号传输、滤波作用。Windows CE.NE 把每个设备表示为文件系统 这样应用程序可以通过这一特殊文件形式来访问驱动程序 主要代码如下 :

```
HANDLE hStr=CreateFile(_T( ADC1: ), GENERIC_READ|GENERIC_WRITE, FILE_SHARE_READ|FILE_SHARE_WRITE,NULL,OPEN_EXISTING,0,0)
```

打开一个设备 ,如同打开一个文件一样 ,设备文件名为 _T(ADC1:)

接下来先调用写函数 写入要读取的管脚号 6:

```
DWORD dwWritten = 0 ,dwBytesRead = ;0
* WriteBuffer = AIN6 写入管脚号 6
WriteFile(hStr, WriteBuffer, 2, &dwWritten, NULL) 写入 AD 管脚号
```

接下来读取 AD 值到 ReadBuffer

```
ReadFile(hStr,ReadBuffer,2,&dwBytesRead,NULL)
读函数得到 10 位的 AD 值到 ReadBuffer 根据这一值计算 AD 的模拟电压 VAD 为  $V_{AD} = \frac{* ReadBuffer}{2^{10} - 1} * 3.3$  进而根据原理图 我们可以得到外部待测点 TEST0 的电压 计算公式为 :
```

$$V_{EST} = \left(\frac{R5+R6}{R6} \right) \left(\frac{R1+R2}{R2} \right) \left(\frac{R4 * 3.3}{R3+R4} - \frac{R1 * V_{AD6}}{R1+R2} \right)$$

5 结束语

基于 Windows CE 的嵌入式产品运用已经越来越广泛 分析了 Win CE AD 驱动程序的开发 通过本文的介绍 读者可以对 Windows CE 驱动程序的结构、编写以及实际应用能有一定的了解。

参考文献

- [1] 傅毓林 等 .Windows CE 的核定制及应用开发 [M] 北京 电子工业出版社 ,2005
- [2] 微软公司 .Microsoft Windows CE 设备驱动程序开发指南 [M] 北京 北京希望电子出版社 ,1999
- [3] Douglas Boling. Microsoft Windows CE 程序设计 [M] 北京 博彦科技发展有限公司 译 北京 北京大学出版社 ,1999

收稿日期 :2006.9.12