

基于 Visual Studio 的虚拟现实与半实物接口设计

邓 飙,刘连伟

(第二炮兵工程大学,西安 710025)

摘要:Virtools 交互式操作中半实物接口的实现方法主要有 2 种。一是编写串口通信类,使用 Virtools SDK 模块开发实现;二是基于 Virtools SDK,通过调用系统应用程序接口函数开发串口通信行为模块实现。2 种方法对开发者使用 C++ 语言编程能力要求很高。针对此问题,提出以 Visual Studio 2010 作为开发平台,使用 C# 开发环境中的串口控件,通过添加 COM 组件中的 Virtools 3D XE Player,实现虚拟现实作品与硬件的交互。实验证明,该方法在实现串口通信的同时,缩短了开发周期。

关键词:Visual Studio;Virtools;虚拟现实;半实物接口;交互式操作

本文引用格式:邓飙,刘连伟. 基于 Visual Studio 的虚拟现实与半实物接口设计[J]. 四川兵工学报,2015(1):114-117.

Citation format:DENG Biao, LIU Lian-wei. Design of Semi-Physical Interface Based on Visual Studio[J]. Journal of Sichuan Ordnance,2015(1):114-117.

中图分类号:TP391.9

文献标识码:A

文章编号:1006-0707(2015)01-0114-04

Design of Semi-Physical Interface Based on Visual Studio

DENG Biao, LIU Lian-wei

(The Second Artillery Engineering University of PLA, Xi'an 710025, China)

Abstract: There are two main methods to realize the semi-physical interface of the Virtools interactive operation. One is that writing serial port communication class with the help of Virtools SDK. The other is that calling the application program interface (API) function of the system to develop the serial port communication building blockings (BB). There is a high requirement on the developers' programming ability of using C++. According to the problem, a new method was put forward. Taking Visual Studio 2010 as an intermediary platform, using serial port control in the C# development environment, Virtools 3D XE Player from the COM components were added to realize the interactive operation between the virtual scene and the physical system. It turns out to be a better way to achieve the designed goal and to shorten the development cycle.

Key words: Visual Studio;Virtools;virtual reality;semi-physical interface;interactive operation

虚拟现实 VR(Virtools Reality)技术又称为“灵境”技术,指用计算机搭建一个逼真的三维感觉世界,使人和计算机很好地“融为一体”,给人以“身临其境”的感觉^[1]。经过近 30 年的发展,虚拟现实技术在许多领域有着广阔的应用前景^[2,3]。虚拟现实的实现需要硬件和软件 2 方面的支持,硬件方面主要包括数据手套、三维鼠标、运动跟踪器、力反馈装

置、语音识别及合成系统等。

Virtools 软件自带了大量的行为模块供开发者使用,但是 Virtools 软件并未提供标准的数据通信模块,不能与虚拟现实仿真系统进行数据通信,因此,虚拟现实仿真系统设计的难点在于半实物接口的设计。朱湘龙^[4],张跃文^[5]利用 Virtools SDK(Software Development Kit),通过调用相应的应

收稿日期:2014-08-06

作者简介:邓飙(1969—),男(苗族),博士,副教授,主要从事系统仿真理论与技术研究;刘连伟(1989—),男,研究生,主要从事虚拟仿真技术研究。

用程序接口函数(API)创建自定义通信模块,实现了实物系统与视景仿真之间的通信,这种方法要求开发者对操作系统的底层函数有很好的理解,并能熟练使用;覃伯明^[6]通过编写串口通信类,利用 Visual Studio2003 编译生成 .dll 通信插件,完成了对串口的打开,读写操作,这种方法对开发者的 C++ 语言使用能力提出了很高要求;傅招国^[7],夏华锦^[8],徐望^[9]利用 Virtools 自带的行为模块(BB)实现了游戏方向盘与仿真系统的交互控制,这种方法较为简单,但连接的外部设备比较单一,工业情况下不适用。

针对此问题,提出以 Visual Studio 2010(以下简称 VS2010)中的 C#开发环境为平台,载入 Virtools 3D XE Player 组件后,利用 VS2010 现有的 SerialPort 串口控件,通过虚拟现实仿真系统(*.cmo 文件)与 VS2010 通信,VS2010 与硬件(单片机)通信,最终实现虚拟现实仿真系统与硬件通信。本文利用 Virtools 自带的实例, Camera Orbit.cmo 与单片机进行信息交互试验,通过单片机上按键控制虚拟场景中石像进行转动或平移运动,以及虚拟场景通过发送消息控制单片机数码管显示不同数字,以此验证本文提出方法的可靠性。逻辑关系如图 1 所示。

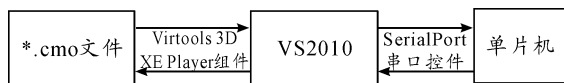


图 1 通信实现逻辑

1 软件平台

1.1 Virtools

Virtools 是一套具备丰富互动行为模块的实时 3D 环境虚拟实景编辑软件。本身自带了 700 多个 Building Block (BB)模块,使用这些模块可以迅速方便地处理丰富和交互性强的 3D 模型。对于某些特殊用途,比如,串口通信的实现,就需要使用 Virtools SDK 开发出新的适用于特殊需求的 BB 模块。SDK 针对 C++ 语言,具有 C++ 面向对象语言的所有特点。

1.2 VS2010

VS2010 是微软公司推出的开发环境,采用拖曳式便能完成软件的开发。简单的操作便可以实现一个界面的生成,支持 C#、C++、VB 等多种语言开发环境。其主要特点有:支持多个监视器、快速浏览代码、调用层次结构项目功能对应等。

2 关键技术及实现

2.1 虚拟现实仿真系统与 VS2010 交互

为减少原型开发周期并减少工作量,本文使用 VS2010 中 C#开发环境与 3D XE Player 组件开发原型的界面。3D XE Player 组件具有以下优点:

1) 支持所有 .NET 开发环境,方便程序开发人员。

2) 不仅提供播放器窗口功能,还包括多种常用方法控件。

C#是可用于创建要运行在 .NET CLR 上的应用程序的语言之一,是微软专门为使用 .NET 平台而创建的。它使用 .NET Framework 代码库的每种功能 其主要特点有语法简单、易于移植到其他操作系统上等。相比于 C++,它是类型安全语言,运行更加稳定。

2.1.1 虚拟现实仿真系统载入

为了在 VS2010 中载入虚拟现实仿真系统,首先要完成 Virtools 3D XE Player 组件的加载。在“工具箱”选项中,依次右键——选择项——COM 组件——Virtools 3D XE Player,点击确定即可加载。

完成组件的加载后,便可以进行仿真系统的载入。以下为 Virtools 3D XE Player 组件载入仿真系统函数原型:

```
public virtual void Load(string file);
```

该函数用于将虚拟场景仿真系统文件,即将 *.cmo 文件载入到 C#开发环境中,其参数 file 有 2 种实现方式,一是填写完整的文件路径名,如“D:\Program Files\3DVIA*.cmo”,这里需要注意的是,文件名之间不能用“\”,否则会出现编译错误;二是填写相对路径,即“Application.StartupPath + “. \ *.cmo””,这里要注意的是,要将待加载的 *.cmo 文件放入到该项目工程中的 Debug 文件夹内。

关键代码如下:

```
private void timer1_Tick(object sender, EventArgs e)
{
    this.axXEPlayer1.Load(Application.StartupPath + ".//Camera Orbit.cmo");
    this.timer1.Stop();
}
```

2.1.2 虚拟现实仿真系统向 VS2010 发送消息

该模块实现的功能是,当虚拟现实仿真系统成功导入到 VS2010 中后,利用 Virtools 软件自带的“Fire Virtools Event”BB 模块实现由虚拟现实仿真系统向 VS2010 发送消息,该模块具有 2 个参数,如图 2 所示。

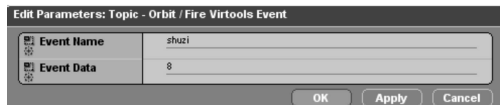


图 2 Fire Virtools Event 参数设置

“Event Name”为发送数据说明,应该简单明了,容易识别,“Event Data”为发送数据内容。

2.1.3 VS2010 接收虚拟现实仿真系统消息

在 VS2010 中,利用 Virtools 3D XE Player 组件中的 VirtoolsEvent 事件接收来自虚拟现实仿真系统的消息,其函数原型如下:

```
private void axXEPlayer1_VirtoolsEvent(object sender, AxXEPlayerActiveX._IXEPlayerEvents_VirtoolsEventEvent e);
```

该函数中需要注意的是第二个参数“e”,其对应于虚拟场景

文件中“Fire Virtools Event”模块中的参数“Event Name”,利用“e.eventData.ToString()”可以获取其所携带的数据,即“Event Data”参数的内容,进而通过对数据内容的判断,做出相应的动作,触发不同的事件。

部分代码如下:

```
private void axXEPlayer1_VirtoolsEvent ( object sender,
AxXEPlayerActiveX._IXEPlayerEvents_VirtoolsEventEvent e)
{
    if (e.eventData.ToString() == "8") //如果仿真系统
发送数字8,则向单片机发送如下信息
    {
        byte[] send1 = new byte[5] { 0x77,0xAA,
0x03,0xbd,0x01 };
        serialPort1. Write(send1,0,5);
    }
    if (e.eventData.ToString() == "9") //如果仿真系统
发送数字9,则向单片机发送如下信息
    {
        byte[] send = new byte[5] { 0x77,0xAA,
0x03,0xbd,0x02 };
        serialPort1. Write(send,0,5);
    }
}
```

2.1.4 VS2010 向虚拟现实仿真系统发送消息

当 VS2010 接收到来自硬件的消息时,需要向虚拟现实仿真系统发送对应消息,虚拟场景中的物体接收到来自 VS2010 的消息后做出相应的动作,借此实现硬件与虚拟场景对象之间的交互。VS2010 中向虚拟现实仿真系统发送消息的函数声明如下:

```
public virtual void BroadcastMessageSingle ( string message)。
```

关键代码如下:

```
private void VT_work ( string text) //VS2010 向仿真系统
发送消息
{
    axXEPlayer1. BroadcastMessageSingle(text);
}
```

2.1.5 虚拟现实仿真系统接收 VS2010 消息

为了实现对 VS2010 所发送消息的接收,虚拟现实仿真系统脚本中需加入多个“Wait Message”BB 模块,该 BB 模块当接收到与设定内容相同的消息时,便触发“out”输出端口。所以,通过对接收内容的设置,及在该模块后添加相应脚本,可以实现不同的动作响应。如图 3 所示为该模块的参数设置对话框,“Message”所接收内容。



图 3 “Wait Message”BB 参数对话框

2.2 VS2010 与硬件交互

单片机具有体积小、功耗低、控制能力强、扩展灵活、微型化和使用方便等优点,目前已渗透到生活的各个领域,几乎很难找到哪个领域没有单片机的踪迹。例如,仪器仪表、家用电器、医用设备、航空航天、专用设备的智能化管理及过程控制领域。单片机与各种硬件之间的交互控制技术已经非常成熟,所以,只要实现 VS2010 与单片机之间的信息交互,再通过单片机与所需控制的外围硬件连接,便可完成 VS2010 对其他外接硬件设备的间接控制。因此,本文只讨论 VS2010 与单片机之间的信息交互实现方法。

2.2.1 VS2010 向单片机发送消息

当 VS2010 接收到来自虚拟现实仿真系统发送的消息时,需要根据消息内容做出相应的判断,并根据判断向单片机发送消息。这一功能实现需要借助 SerialPort 串口控件的“Write”函数,其函数声明如下所示:

```
public void Write(byte[] buffer,int offset,int count);该函
数的“buffer”参数表示一个字节数组,“offset”参数表示从
“buffer”数组中第几个字节开始写入串口,“count”参数表示
需要写入的字节数量。
```

其关键代码参看 2.1.3 小节。

2.2.2 VS2010 接收单片机消息

此功能模块实现的是,VS2010 作为单片机与虚拟现实仿真系统的中介平台对单片机发送的消息进行接收,这一功能实现需要借助 SerialPort 串口控件中的“DataReceived”事件,及“Read”函数(与“Write”函数类似,不再赘述)。需要注意的是,串口控件在工作时为保证能实时接收和发送消息,会在主线程外开辟一个新的线程。所以,当在“DataReceived”函数事件中向仿真系统发送来自单片机的消息时,需要进行委托,才能进行跨线程参数传递,否则编译会出现错误。

关键代码如下:

```
private void serialPort1_DataReceived ( object sender, Sys-
tem. IO. Ports. SerialDataReceivedEventArgs e)
{
    byte[] receive;
    s = null;
    int num = serialPort1. BytesToRead; //确定单
片机发送给 VS2010 数据的字节数
    if (num > 0)
    {
        receive = new byte[num];
        serialPort1. Read(receive,0,num); //读取单
片机发送数据
        for (int i = 0; i < num; i++)
        {
            s += receive[i]. ToString("X2");
        }
        if (s == "77AA03BDEE22") //根据单
片机发送数据,向仿真系统进行跨线程对应参数传递
```

