

doi: 10.3969/j.issn.1006-1576.2011.09.021

基于 IVI 驱动模式的仪器信号驱动器设计

马羚¹, 吕晓峰², 马登武²

(1. 海军航空工程学院研究生管理大队, 山东 烟台 264001;

2. 海军航空工程学院兵器科学与技术系, 山东 烟台 264001)

摘要: 针对测试平台中仪器的互换性问题, 提出一种基于 IVI 驱动模式的仪器信号驱动器设计方法。通过分析测试软件平台的架构, 建立基于 IVI(interchangeable virtual instrument)驱动模式的仪器信号驱动器, 研究驱动器的结构和接口, 并采用 COM(component object model)组件技术完成了仪器信号驱动的设计。试验结果证明: 该设计实现了对仪器驱动的面向信号调用, 提高了仪器的可互换性。

关键词: IVI; 面向信号; 仪器驱动器; ATS

中图分类号: TP216 **文献标志码:** A

Design of Signal-Oriented Instrument Driver Based on IVI Driving Mode

Ma Ling¹, Lu Xiaofeng², Ma Dengwu²

(1. Administrant Brigade of Postgraduate, Naval Aeronautical & Astronautical University, Yantai 264001, China;

2. Dept. of Ordnance Science & Technology, Naval Aeronautical & Astronautical University, Yantai 264001, China)

Abstract: For improving the interchangeability of automatic test system (ATS), a signal-oriented instrument driver is designed based on IVI driving mode. In this paper, after analyzed the software framework of ATS, the signal-oriented instrument driver based on interchangeable virtual instrument (IVI) driving mode is established, then the structure and interface of the driver is studied and the signal-oriented instrument driver is designed by component object model (COM) technique. Experiment result demonstrates that the design can realize the signal-oriented use for instrument driver and improve the interchangeability of ATS.

Keywords: IVI; signal-oriented; instrument driver; ATS

0 引言

“面向仪器”的 IVI-C、IVI-COM 和 IVI-MSS 经过长足发展已经比较成熟, 能部分地满足开发人员的互换需求。但由于仪器种类繁多, 每种仪器控制方式多样, 新仪器、新功能不断涌现, 难以实现完全的仪器可互换性。“面向信号”的 IVI-Signal 为实现仪器的完全互换提供了完美的解决方案, 但迟迟未发布其正式标准。笔者参考 IVI-Signal 思想, 结合 IEEE1641 标准^[1], 提出了一种基于 IVI 驱动模式的仪器信号驱动器设计方法。

1 ATS 软件平台架构

软件平台^[2]是 ATS 的核心, 主要由仪器信号驱动、接口框架、资源分配器和信号 4 组件组成, 如图 1。

1) 仪器信号驱动: 测试功能的实现者, 通过封装底层仪器驱动函数, 完成具体仪器信号测试功能。

2) 接口框架: 定义面向信号仪器驱动各接口方法和属性, 供其他组件调用。

3) 资源管理器: 应用程序和具体仪器信号驱动的中间层, 用于检索仪器信号能力文件(XML 文

件), 查找合适的仪器, 创建具体仪器信号驱动。

4) 信号组件^[3]: 每个信号组件对应一种信号, 其功能是存储客户程序输入的信号信息, 供具体仪器信号驱动调用; 存储测试结果, 供应用程序调用。

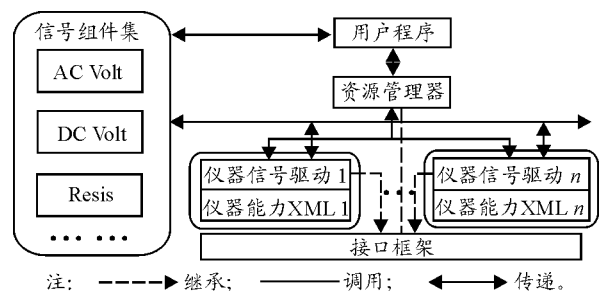


图 1 ATS 软件平台架构

2 仪器信号驱动器设计

2.1 结构设计

仪器信号驱动支持多个物理信号的产生或测量工作, 物理信号既可属于相同种类, 又可属于不同的种类。为了支持多信号同时控制机制, 仪器信号驱动必须支持多个信号控制接口, 每个信号接口控制一种信号。为了达到上述要求, 可采取以下方式: 每个仪器信号驱动是一个 COM 组件^[4], 具有多个

收稿日期: 2011-06-10; 修回日期: 2011-06-28

作者简介: 马羚(1987—), 男, 四川人, 硕士, 从事航空武器自动测试研究。

接口，每个接口实现不同的信号控制。

仪器信号驱动包括信号角色管理接口 (IDriver) 和信号角色接口 (IACVoltSensor、IDCCurrentSource) 等。每台仪器对应一个信号角色管理接口，该接口用于获取信号信息(如电阻、交流电压信号)、仪器扮演的信号角色，完成信号角色接口的调用；配置仪器测量或触发状态；完成信号角色接口的释放^[5]。信号角色接口根据封装厂家提供的仪器驱动程序，该接口通过分解仪器在测试过程中扮演的角色和仪器的信号能力得到 2 组数据，将两者组合得到一组涵盖仪器信号能力和角色能力的驱动程序。因此，将仪器信号角色与仪器信号能力两者组合即可得到仪器信号角色的接口类型。例如：IACVoltSensor 接口(交流电压测量接口)、IACSource(交流信号源接口)，图 2 为仪器信号驱动结构图。

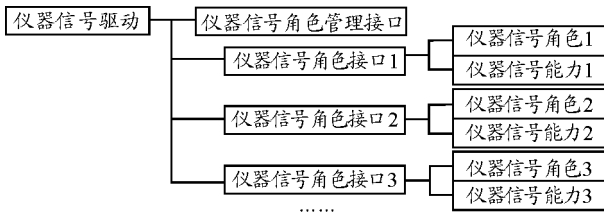


图 2 仪器信号驱动结构

IVI-Signal^[6]规范定义了 5 类仪器的信号角色，

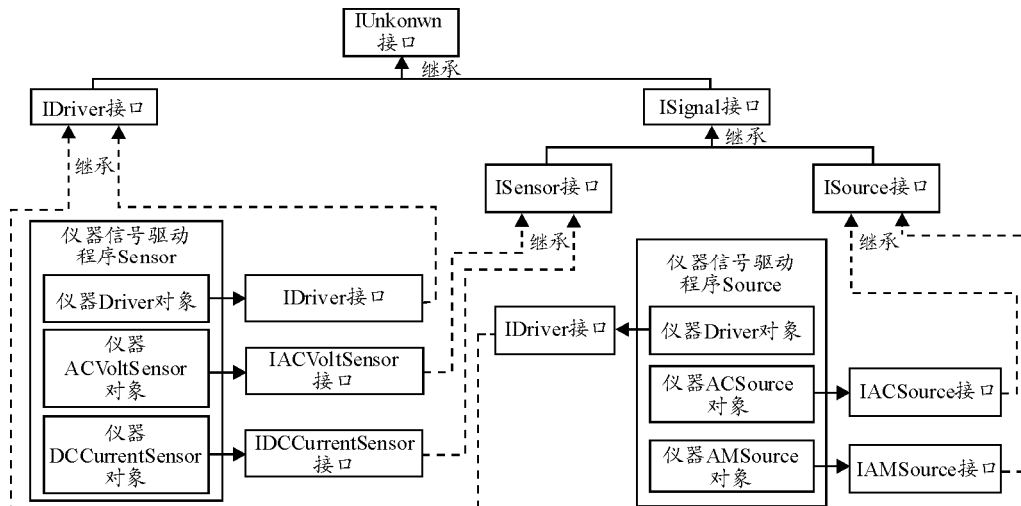


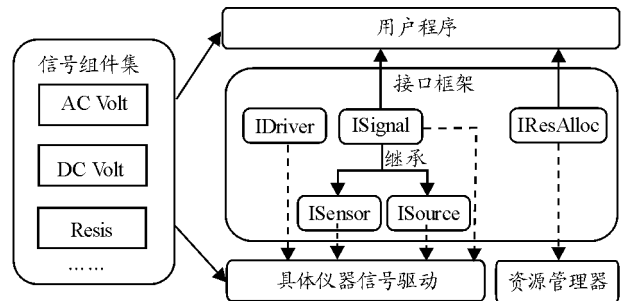
图 4 仪器信号驱动接口继承方式

具体仪器信号驱动通过创建继承于 IDriver、ISignal、ISource 和 ISensor 接口，实现了接口中定义的方法。接口继承与 C++ 中类的继承不同。首先类继承不仅是说明继承，也是实现继承，即派生类可以继承基类的函数实现，而接口继承只是说明继承，即派生的接口只继承了基接口的成员函数说明，并没有继承基接口的实现，因为接口定义不包括函数实现部分。其次，类继承允许多重继承，一个派

包括 Source、Load、Sensor、Connect 和 Switch。笔者研究的仪器符合 Source、Load 和 Sensor 3 种角色。由于 Source 控制仪器产生信号或激励，Load 用于信号转换并输出，从输出角度来说两者功能类似；同时 IVI-Signal 规范对这 2 个角色定义的接口方法一致，可将两者合并为 Source。

2.2 接口设计

仪器信号驱动接口设计包括 2 方面内容，一方面是接口框架的建立，如图 3，另一方面是仪器信号驱动接口的继承方式，如图 4。其中信号驱动接口框架定义了接口的方法和属性，仪器信号驱动接口继承于接口框架并完成接口方法的具体实现。



注：-----> 组件继承接口；——> 接口间继承；——> 用户程序接口。

图 3 接口框架

生类可以有多个基类，但接口继承只允许单继承，不允许多重继承^[7]。

2.2.1 根接口设计

由图 4 可见，根接口包括 IDriver 接口和 ISignal 接口，这 2 个接口是从 IUnknown 接口继承得到的双重接口。使用双重接口可以使 COM 对象易于开发、易于修改，便于在大多数开发环境下使用。

IDriver 接口主要有以下功能：配置仪器初始状

态; 通过获取信号信息(如电阻、直流电压信号)、仪器扮演的信号角色, 完成对仪器信号驱动信号角色接口的调用; 完成信号角色接口的释放以及仪器的复位与关闭。该接口方法包括 Init、CreateSignal、Close、Reset 和 FreeSignal。以 Init 为例, 其方法和参数定义如下:

```
HRESULT Init(CAServerObjHandle objHandle,
char *LogicName, char *Address, char *Capability)
```

功能: 根据逻辑名、物理地址和信号能力信息完成仪器初始化。

参数: objHandle, 仪器 Driver 对象句柄; LogicName, 逻辑名; Address, 物理地址; Capability, 仪器能力信息。

ISignal 接口是所有信号角色接口的根接口, 主要完成复位仪器信号驱动和初始化仪器信号驱动功能。该接口方法包括 Reset 和 Setup。以 Setup 为例, 其方法和参数定义如下:

```
HRESULT Setup(CAServerObjHandle
objHandle, IUnknown *Objhandle)
```

功能: 初始化具体仪器驱动, 使仪器动作到相应的控制状态。

参数: objHandle, 仪器驱动信号对象句柄; Objhandle, 用户输入的信号组件接口指针。

2.2.2 角色接口设计

角色接口包括 ISensor 接口和 ISource 接口, ISensor 接口和 ISource 接口继承于 ISignal 接口。ISensor 接口继承了 ISignal 接口 2 种方法: Setup、Reset, 自定义了 Initiate、Fetch 方法, ISource 接口继承了 ISignal 接口 Setup、Reset 方法, 完成信号触发、加载功能。以 ISensor 接口自定义的 Initiate 为例, 其方法和参数定义如下:

```
HRESULT Initiate(CAServerObjHandle
objHandle, char *Paramlist)
```

功能: 根据参数列表使仪器完成测量功能。

参数: objHandle, 仪器驱动信号对象句柄; Paramlist, 参数列表。

2.3 接口开发

接口框架的实现是通过对根接口、角色接口创建相应的 COM 组件, 完成接口方法的定义。接口包含了一组函数的数据结构, 通过这组数据结构, 应用程序可以调用 COM 组件对象的功能。

应用程序用 1 个指向接口数据结构的指针来调用接口成员函数, 如图 5, 接口指针实际上又指向另一个指针, 这第 2 个指针指向一组函数, 称为接

口函数表, 也称作虚函数表(virtual function table, vtable), 指向 vtable 的指针为 pVtable。接口函数表中每一项为 4 个字节长的函数指针, 每个函数指针与对象的具体实现连接起来。通过这种方式, 应用程序只要获得了接口指针, 就可以调用到对象的实际功能。

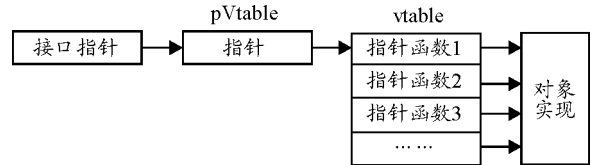


图 5 接口结构

3 仪器信号驱动的实现

笔者以 Sensor 类数字万用表 HP34401^[8]为例, 在 Lab Window/CVI 8.0 开发环境下, 通过编写测试程序, 调用资源分配器组件、信号组件、仪器信号驱动组件完成仪器信号驱动的实现。HP34401 信号驱动流程如图 6。

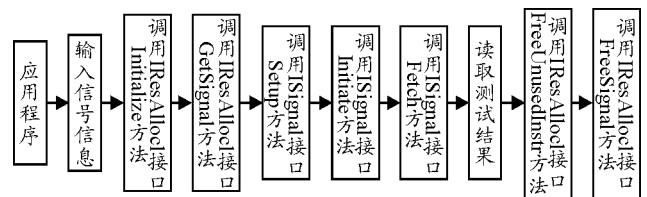


图 6 HP34401 信号驱动流程

其信号驱动步骤:

- 1) 应用程序输入信号信息, 信号组件存储这些信号信息, 供具体仪器信号驱动调用。
- 2) 应用程序调用资源管理器 Initialize 方法, 应用程序已经指定仪器, 该方法在此处可以不必调用。
- 3) 应用程序调用资源分配器 GetSignal 方法, 创建具体仪器信号驱动, 进行信号分配。
- 4) 应用程序调用 ISignal 接口 Setup 方法配置具体仪器信号驱动。
- 5) 应用程序调用 ISignal 接口 Initiate 方法根据参数需求使仪器完成测量功能。
- 6) 应用程序调用 ISignal 接口 Fetch 方法, 将测量结果存储到信号组件中。
- 7) 应用程序调用相应的信号组件, 返回测试结果。
- 8) 应用程序调用 IResAlloc 接口 FreeUnusedInstr 方法释放仪器。
- 9) 应用程序调用 IResAlloc 接口 FreeSignal 方法释放信号。

4 结束语

应用结果表明: 该驱动器将面向信号技术与 COM 技术结合, 实现了对仪器驱动的面向信号调用, 使测试程序集与测试资源分离, 提高了测试程序的可移植性和测试资源的可互换性。

参考文献:

[1] IEEE. IEEE1641-2007 Standard for Signal&Test Definition(STD)[S]. 2007.

[2] 沈震, 戴英侠, 杨江平. 自动测试设备软件平台通用性的研究与设计[J]. 计算机工程与应用, 2005, 3(9): 229-232.

[3] 戴豪民, 牛双诚, 张光轶. 面向信号自动测试系统信号组件的设计与实现[J]. 现代电子技术, 2009(4): 60-62.

[4] 潘爱民. COM 原理与应用[M]. 北京: 清华大学出版社,

2005.

[5] 王学浩, 岳瑞华, 徐中英, 等. 导弹自动测试系统性能评价指标及体系[J]. 四川兵工学报, 2010, 31(9): 37.

[6] IVI Foundation. IVI-3.11: signal interface specification [EB/OL]. Http://www.ivifoundation.org, 2000, 10.

[7] 刘宗田, 袁兆山, 潘秋林. C++编程思想[M]. 北京: 机械工业出版社, 2006.

[8] Agilent China. Agilent Multimeter 34401a User manual[DB/OL]. Http://www.Home.agilent.com, 2006, 06.

(上接第 67 页)

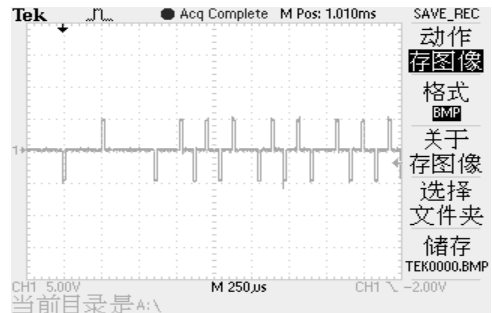
```

if(flag!=0x01){
    asm("clrc INTM");
    return;}
else {*EVIFRC=*EVIFRC|0x01;
    pre_cnt=load();//保存计数值
    fnp_flag=1; }}
else {flag=*EVIFRC&0x04;
    if(flag!=0x04){
        asm("clrc INTM");
        return;}
    else {*EVIFRC=*EVIFRC|0x04;
        ur_cnt=load();//保存计数值
        if(cur_cnt-pre_cnt<0){
            t=cur_cnt+0xffff; }
        if(cur_cnt-pre_cnt>=75)
        &&(cur_cnt-pre_cnt<=100)){
            find_sy=1;}}//找到同步头
    else {find_sy=0; //清除找到同步头标志, 为找
    下一帧同步头作准备
    //使用与找同步类似的方法, 每个一个时钟读
    取数据, 保存数据并进行奇检验。以下程序省
    略……}}

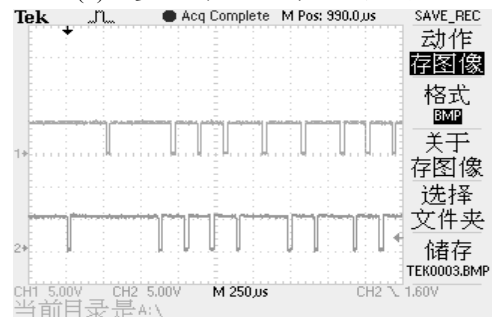
```

4 实验调试

在硬件电路调试时, 采用多臂并径仪作为曼彻斯特 II 码信号发生源, 对该部分所涉及硬件电路单元的功能测试。图 7(a)给出了自制信号发生器产生的曼彻斯特 II 码信号波形。图 7(b)给出了曼彻斯特 II 码信号经过硬件转换单元后的波形, 通道 1 当中窄脉冲信号的位置对应转换前, 正窄脉冲信号所处的位置, 通道 2 当中窄脉冲信号的位置对应转换前, 负窄脉冲信号所处的位置。这 2 路处理后的窄脉冲信号, 通过电平转换模块送入 F2812 捕获单元所对应的芯片管脚之上, 利用软件完成对曼彻斯特 II 码信号的解析操作。



(a) 曼码信号发生器产生的波形



(b) 曼彻斯特 II 码处理后的波形

图 7 曼彻斯特 II 码调试波形

5 结论

实验调试结果表明: 该系统的整体设计方案合理, 系统硬件设计和系统软件算法设计正确, 能够准确对曼彻斯特 II 码进行解析, 提高了解析系统的灵活性和准确性, 降低的系统硬件设计成本, 可以满足生产测井过程中常规测井作业的需求。

参考文献:

[1] 高嵩, 何宁, 王国琿, 等. 曼彻斯特码多路数据传输系统[J]. 西安工业学院学报, 2003, 23(4): 294-298.

[2] 高嵩, 王国琿, 潘泉. USB 总线在便携式数控测井系统中的应用[J]. 弹箭与制导学报, 2005, 25(4): 437-439.

[3] 袁晓伟, 高嵩, 陈超波, 等. 基于 DSP 便携式数控测井系统的设计[J]. 电子设计工程, 2009, 17(11): 27-29.

[4] 童诗白, 华成英, 等. 模拟电子技术基础[M]. 3 版. 北京: 高等教育出版社, 2002: 173-214.

[5] 戴明桢, 周建江. TMS320C54x DSP 结构、原理及应用[M]. 北京: 清华大学出版社, 2007: 177-221.

[6] 胡尔佩. 数控测井系统技术基础[M]. 北京: 石油工业出版社, 1995(2): 7-10.