
虚拟仪器 3.0



V1.0.0 版

《虚拟仪器 3.0》编写委员会 编著

虚拟仪器 3.0

本书编委（排名不分先后）：

景博（空军工程大学）
陆芳（空军工程大学）
兰京川（电子科技大学）
陈晓龙（西安电子科技大学）
周强（北京航空航天大学）
陈大庞（简仪科技）
吴珂珮（简仪科技）
刘隽（简仪科技）
邵天宇（简仪科技）
宋文波（简仪科技）
张喜庆（聚星仪器）

责任编辑：

徐云（简仪科技）

前言

各位读者，

虚拟仪器技术在过去的三十多年中取得了令人瞩目的成就。诸如合成仪器，柔性测试，软件仪器等各种各样的技术名词都描绘了软件是虚拟仪器核心这一事实。软件技术的发展又是和计算机硬件技术的发展紧密相连的。PC 机的诞生使得工程师能够用 BASIC 等语言编制一些软件；图形操作系统的诞生造就了 LabVIEW 这样有代表性的图形编程仪器软件。时代的脚步并没有停留在 BASIC，C/C++ 等过程编程语言。面向对象的跨平台编程语言 Java，C#，Python 等极大提高了通用编程的效率，已经成为无可争议的主流编程软件。把这些语言引入到测控专业是必然的。这些新的软件技术加上开源社区把虚拟仪器技术提升到一个崭新的阶段：虚拟仪器 3.0。

虚拟仪器技术 3.0 的另一个特点是编程语言不再是某一家厂商的专有语言。这样工程师只要掌握一门主流编程语言就能从容应付迅速发展的时代变化。大大地提高了效率和减少了不必要的重复劳动。

简仪科技推出的锐视测控平台正是顺应通用软件技术的发展趋势而诞生的。我们首选 C# 是因为考虑到微软 Windows 在科技界的广泛使用；我们使用 Visual Studio 是因为它是一个非常好用的开发工具；我们接纳开源的理念是因为历史证明开源的系统一旦站住脚就无往而不胜。

这本书是简仪和一些早期看好开源测试软件的老师一起在很短的时间里写出来的。严格来说现在还只是一个初稿。我们会不断地改善，丰富这本书的内容。众人拾柴火焰高，我们欢迎更多的老师和读者加入到完善这本书的队伍里来。让这本书成为学习虚拟仪器技术最有益的参考书，同时也给需要教学的老师提供一些帮助。

陈大庞 博士
简仪科技

目录

- 1** 绪论
- 3** 传感器与信号调理
- 18** 总线接口技术
- 50** 虚拟仪器 3.0 的驱动设计
- 55** 虚拟仪器 3.0 的界面设计
- 63** 虚拟仪器 3.0 的分析工具
- 66** 虚拟仪器 3.0 的应用领域

虚拟仪器 3.0 绪论

虚拟仪器技术的定义和发展

PC 在过去 20 年的迅速普及促成了测试、测量和自动化仪器技术的革命。PC 的普及推动了虚拟仪器技术概念的产生，为需要提高生产力，准确性和性能的工程师和科学家提供了几个好处。

新的以软件为中心的虚拟仪器系统为用户提供了新技术并大幅降低了生产成本。通过虚拟仪器，工程师和科学家们可以精确地（用户定义）构建满足其需求的测量和自动化系统，而不是受传统固定功能仪器（供应商定义）的限制。

测试测量是产品质量保障的关键部分，而虚拟仪器技术中的软件部分则在测试过程中集合了数据的采集与测量、测控过程中的控制决策和输出以及数据的处理、分析和管理等功能，是测试测量过程的重中之重。“软件就是仪器”。

虚拟仪器软件的发展经历了文本语言编程 (Basic,VB.NET) → 图形编程 (LabVIEW, VEE) 的发展历程。目前主流的测控软件采用图形编程的方法 (LabVIEW)，虽然图形化编程支持复杂和通用的图形界面，并且在实时性以及分布式 IO 等方面具有优势，但在进行大系统开发时的易用性不足，同时维护起来也比较麻烦，与此同时图形编程不是开源，难以进行跨平台开发，这就使得图形编程的测控软件不仅费用成本较高，且调试麻烦，效率低下。

为了有效地克服当前的软件所存在的不足之处，简仪科技提出并设计了一种面向对象的开源测控软件——锐视开发软件，它的诞生则开启了使用面向对象编程 (OOP) 技术的最新文本语言来控制仪器的时代，完成了文本 (Basic,VB.NET) → 图形 (LabVIEW,VEE) → 文本 (C#) 的循环。锐视开发软件不仅实现了开源，同时在开发过程中以类象作为最小编程单位，这就大大方便了仪器编程，而且克服了图形编程测控软件在跨平台开发、产品维护、费用以及工作效率等方面的不足。同时也可为 PXI 联盟厂商带来了更大的设计灵活性和开源共享同一个软件平台的可能性，从而可以更好的赢得更多的客户，一同开拓市场。

虚拟仪器技术发展的趋势

锐视开发软件是由简仪科技自主开发的免费、强大、易用、开源的测控系统开发软件，其借助于 Microsoft.NET 和 Microsoft VisualStudio 开发环境的强大技术支撑，锐视开发软件的完成和发布使得锐视测控平台（包括锐视开发软件 + 简仪科技模块测量硬件产品线）将成为业界第一个功能强大、且完全开源的测控系统专业开发平台。锐视测控平台提供基于 Visual C# 语言和 x86 架构的一系列软硬件解决方案，是的仅具有基本 C 语言基础的测控工程师也可以轻松快速地开发出功能强大、界面专业、易于维护和扩展的测控系统，从而大大提高测控系统的开发效率，实现软硬件之间的无缝链接。

锐视开发软件是简仪潜心研发全球首个开源测控软件，其核心技术是全新的 MACOs: Measurement Automation Class Object(s)，面向对象的测控软件技术架构，中文名称：测控类象。锐视开发软件是以类 (Class)、对象 (Object) 为应用程序的最小编程单位，且所有的类象都是开放型的，都具有明确的方法 (Methods) 和属性 (Properties) 定义。测控领域的工程师可利用锐视软件开发出高效、可靠、开放、可扩展的新一代测控系统。

由锐视开发软件和符合 MACOs 驱动架构的模块仪器硬件就组成了虚拟仪器 3.0。

虚拟仪器的基本要素

一个虚拟仪器由一个工业标准的计算机或工作站组成，这些计算机或工作站配备有强大的应用软件，高性价比的硬件，如插件板和驱动程序软件，它们共同执行传统仪器的功能。虚拟仪器代表着从传统硬件为主的测量系统到以软件为中心的测量系统的根本性转变。以软件为主的测量系统充分利用了常用台式计算机和工作平台的计算、显示和互联网等诸多用于提高工作效率的强大功能。虽然 PC 机和集成电路技术在过去的 20 年里有着巨大的发展和提高，但是，软件才是在功能强大的硬件基础上创建虚拟仪器系统的真正关键所在。虚拟仪器的基本要素包括了：1，可自定义功能的仪器软件；2，和软件无缝结合模块硬件和仪器。

虚拟仪器 3.0 绪论

软件是虚拟仪器最重要的组成部分。通过正确的软件工具，工程师和科学家可以通过设计和集成特定过程所需的例程来有效地创建自己的应用程序。他们还可以创建一个最适合应用程序的用户界面以及与之交互的用户界面，还可以自己定义应用程序如何以及何时从硬件设备中采集到数据，如何处理、分析并储存数据，及如何显示结果。

有了功能强大的软件帮助，您就可以为仪器设备设置智能化的决策功能，以便在被测信号突然发生变化或需要更高或更低的处理功能时即时做出反应。

软件所能提供的一个重要优势就是模块化。在处理一个大项目时，工程师和科学家们通常会根据不同功能将其分成几个单元。分割之后的子任务更加容易处理，容易进行测试，也减少了会引起意外行为的依赖关系。您可以设计不同的虚拟仪器来执行各个子任务，然后再将它们集成到一个完整的系统中执行大型任务。而让您能够如此简单地实现任务划分的根本原因在于软件的构架。

虚拟仪器 1.0

虚拟仪器 1.0 诞生于计算机辅助测试 (CAT) 开始实践的 80 年代，当时主流的方式是通过 HP Basic 语言 + GPIB 仪器控制接口卡，通过个人计算机来控制台式仪器设备。测试工程师通过传统的文本编程实现从台式仪器中获得数据和结果显示。随着 APPLE II 等带有开放插卡扩展的计算机的出现，有工程师开发出数据采集卡，软件方面继续沿用 HP Basic 编写相应的测试模块。

虚拟仪器 2.0

随着计算机图形处理能力的不断加强，NI 美国国家仪器提出了 LabVIEW 图形化的测试开发软件，提高了工程人员开发测试计算软件的效率和易用性。虚拟仪器 2.0 时代几乎被 NI 一家所垄断，LabVIEW 几乎成了虚拟仪器的代名词，期间是德科技（原 Agilent 电子测量部）的 VEE 同样是图形化测试开发软件，但并没有被广泛的应用。随着 LabVIEW 的不断发展，由 NI 最初定义的 PXI 模块化仪器也迅速发展。目前 NI 的 LabVIEW 在 PXI 模块仪器领域已经占领了超过一半的市场。

虚拟仪器 3.0

由于测试测量行业的发展相对封闭和缓慢，相比当今的工程技术迅速发展，有很多优秀的技术并没有得到足够多的了解和重视，特别是在软件工程领域。现在的软件工程已经全面转向面向对象的编程思想，相比较测控界的软件工具和编程方法还停留在面向过程，LabVIEW 就是面向过程软件的典型代表。面向过程能够快速解决小项目开发，但是在中型和大型复杂的定制化测试项目，面向过程的图形编程的各种弊病显现：可读性差，可维护性低，执行效率低，多线程实现难。相反，以 C# 为代表的微软主推的面向对象的语言快速发展，可以完美解决以上问题，其智能文本编程的编译环境易学易用，同时所有的编程逻辑和结构都可以进行面向对象的设计，使得代码重用，团队项目协作变得轻松自如。由于 Visual Studio/C#/.NET 是通用开发环境 (IDE)，大量的应用并不在测试测量行业，又由于有一些测试测量行当专用的工具没有在此 IDE 中，这就造成了 Visual Studio/C#/.NET IDE 没有成为测试仪器主流软件。尽管如此，很多用户都已自发地发现了 Visual Studio/C#/.NET 的潜力，成功地开发了数不尽的测试测量应用。

简仪科技作为一个新成立的中国自主知识产权的测试公司，敏锐地察觉到 Visual Studio/C#/.NET 和开源社区的巨大潜力，在全球第一个创造性提出了直接用微软 C# 语言和 .NET 技术来直接搭建一个完全符合面向对象思维的测控软件平台：锐视开发软件 — SeeSharp®。锐视开发软件 SeeSharp® 把成熟的 Visual Studio/C#/.NET 技术和开源社区的丰富内容进行系统的整理，有条不紊地引入到测控行业，将彻底改变 LabVIEW 垄断通用测试软件现状，也将成为虚拟仪器 3.0 的代表。

传感器与信号调理

信号的类型及其检测

在虚拟仪器中，通常是采用传感器将被测的物理量转换为信号量，而后进行采集的。按照传感器输出信号的不同，可以分为以下几类：

(1) 电阻测量

传感器输出为电阻类的信号。这类传感器包括：常规空气温度传感器，气候观测的气温传感器、地温传感器等。对于虚拟仪器的测量通道，常常采用电阻测量方式。

(2) 电流测量

传感器输出为电流类的信号。这类传感器典型的包括：超声波蒸发传感器；采集的方法通常采用电流测量方式。

(3) 电压测量

传感器输出为电压信号。这类传感器包括：湿度传感器、红外地温传感器的腔体温度；采集的方法为电压测量方式。

(4) 小信号差分电压测量

传感器输出为小信号差分电压信号。这类传感器包括：辐射传感器、红外地温传感器的目标温度；采集的方法通常采用差分电压测量方式。

(5) 脉冲频率测量

传感器输出为脉冲频率信号。这类传感器包括：风速传感器、称重雨量传感器等。采集的方法通常采用脉冲频率测量方式。

(6) 通断计数测量

传感器输出为通断信号。这类传感器包括：雨量传感器、机箱门开关检测；采集通道采用计数测量方式。

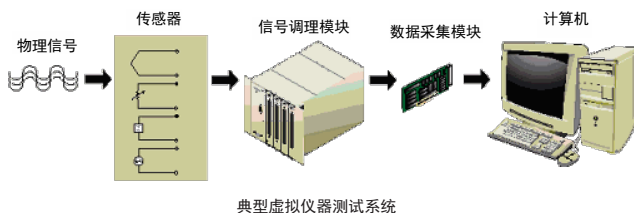
(7) 数字电平测量

传感器输出为数字电平信号。这类传感器包括：格雷码风向传感器；采集的通道采用数字电平测量方式。

信号调理概述

(1) 信号调理的概念与定义

所谓信号调理就是对被测的输入和测试设备本身的输出信号（包括：数字信号和模拟量信号）进行相关的特性变换、幅值衰减、放大、隔离、驱动、滤波等操作，使其相互匹配。从而使得测试设备的可靠性及性能得到有效地改善。



(2) 信号调理的重要性

信号调理的重要性的体现在以下几个方面：

- ① 信号调理往往是测试设备与被测产品（DUT）的实际物理接口；
- ② 信号调理通过滤波、隔离等措施消除噪声；
- ③ 信号调理的驱动能力直接影响测试精度；
- ④ 信号调理精度影响测试误差；
- ⑤ 信号调理影响整个测试设备的构型；
- ⑥ 信号调理可以保护被测产品；
- ⑦ 信号调理保护测试设备。

(3) 信号调理的发展趋势

伴随着技术的发展，信号调理正向集成化、通用化、模块化、智能化、可编程方向迅速发展，有诸多公司都推出了系列化、模块化的信号调理产品，值得选用和借鉴。

(4) 调理基本技术概述

① 隔离

隔离又称电路隔离，其主要目的是通过隔离元器件把噪声干扰的路径切断，并且保护被测产品和测试设备。

传感器与信号调理

通俗的说，就是测试设备与被测产品不共“地”。电路隔离主要有：模拟电路的隔离、数字电路的隔离、数字电路与模拟电路之间的隔离。常用隔离的方法包括：变压器隔离法、脉冲变压器隔离法、继电器隔离法、光电耦合器隔离法、直流电压隔离法、线性隔离放大器隔离法、光纤隔离法、A/D 转换器隔离法等。

其中最为关键之一的是电源隔离，电源隔离又可以细分为：交流电源隔离和直流电源隔离。

◆ 交流供电隔离

由于交流电网中存在着大量的谐波，雷击浪涌，高频干扰等噪声，所以对用交流电源供电的控制装置和电子电气设备，都应采取抑制来自交流电源干扰的措施。采用电源隔离变压器，可以有效地抑制窜入交流电源中的噪声干扰。

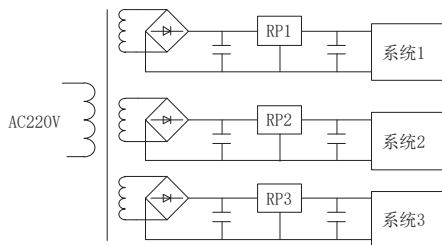
一般变压器原、副绕组之间虽也有隔离电路的作用，但在频率较高的情况下，两绕组之间的电容仍会使两侧电路之间出现静电干扰。

为避免这种干扰，隔离变压器的原、副绕组一般分置于不同的芯柱上，以减小两者之间的电容；也有采用原、副绕组同心放置的，但在绕组之间加置静电屏蔽，以获得高的抗干扰特性。

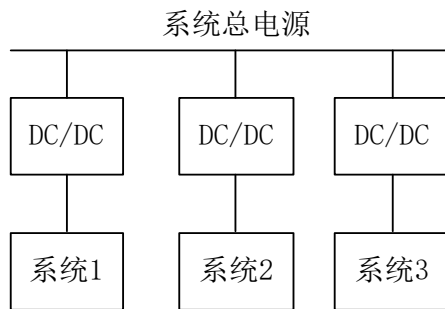
静电屏蔽就是在原、副绕组之间设置一片不闭合的铜片或非磁性导电纸，称为屏蔽层。铜片或非磁性导电纸用导线连接于外壳。有时为了取得更好的屏蔽效果，在整个变压器，还罩一个屏蔽外壳。对绕组的引出线端子也加屏蔽，以防止其他外来的电磁干扰。这样可使原、副绕组之间主要只剩磁的耦合，而其间的等值分布电容可小于 0.01pF，从而大大减小原、副绕组间的电容电流，有效地抑制来自电源以及其他电路的各种干扰。

◆ 直流供电隔离

当控制装置和电子电气设备的内部子系统之间需要相互隔离时，它们各自的直流供电电源间也应该相互隔离，其隔离方式如下：第一种是在交流侧使用隔离变压器，第二种是使用直流电压隔离器（即 DC/DC 变换器），如下图所示。

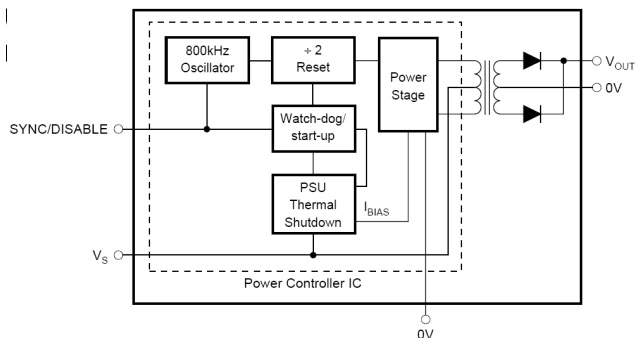


直流变压器隔离

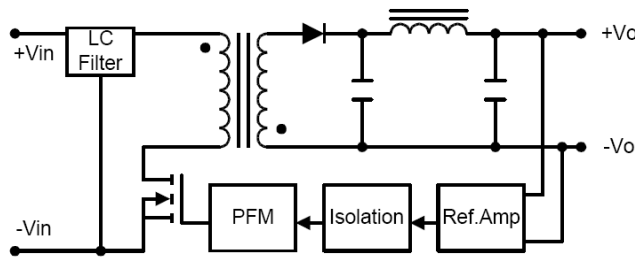


直流 DC/DC 隔离

由于 DC/DC 隔离体积小、集成度高，所以应用广泛。常用的 DC/DC 模块包括：DCP 020515DP、MINMAX 模块、金升阳模块等等。这类模块的内部基本上采用变压器隔离，而要利用变压器隔离，必然采用 DC 变 AC 后输入到变压器的原边，而后在变压器的副边的到 AC 再变换为 DC。例如：DCP 020515DP、MINMAX 稳压 DC/DC 模块内部原理如下图所示。



DCP 020515DP 内部结构



MINMAX 稳压 DC/DC 模块内部结构

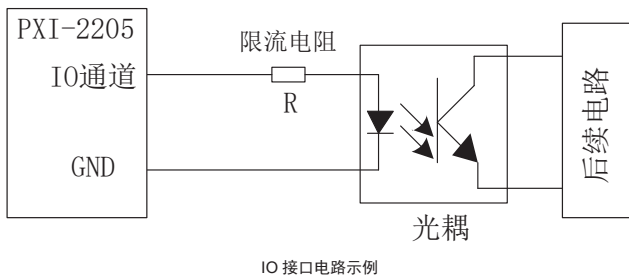
② 驱动能力

◆ 数字量

对于数字量来说，其驱动能力可以表征为扇出系数。驱动能力强的芯片其驱动电流可达几十~一百多 mA。

传感器与信号调理

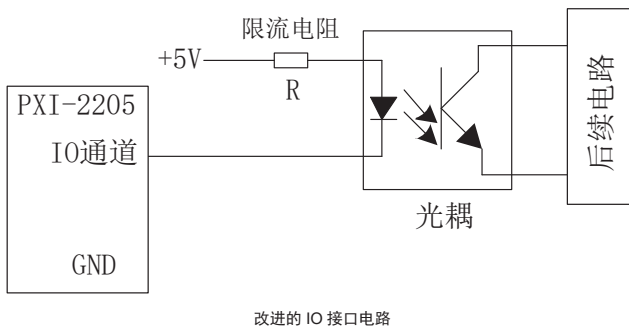
例如在某工程项目中，测试设备中调理板卡的某通道接口采用光耦接收电路，调试中发现 PXI-2205 板卡的 IO 输出通道却无法驱动光耦接收电路，如下图所示。



查阅技术手册，PXI-2205 板卡 IO 通道输出特性如下：

- 低 (Low): $V_{OL}=0.5V \text{ max}$, $I_{OL}=8mA \text{ max}$
- 高 (High): $V_{OH}=2.7V \text{ min}$, $I_{OH}=400\mu A \text{ max}$

从技术特性可以看出，其输出高电平的驱动能力过低，无法直接驱动光耦电流导通。我们可以反过来用低电平来驱动光耦工作，因为 I_{OL} 为 $8mA$ 足以使光耦导通。



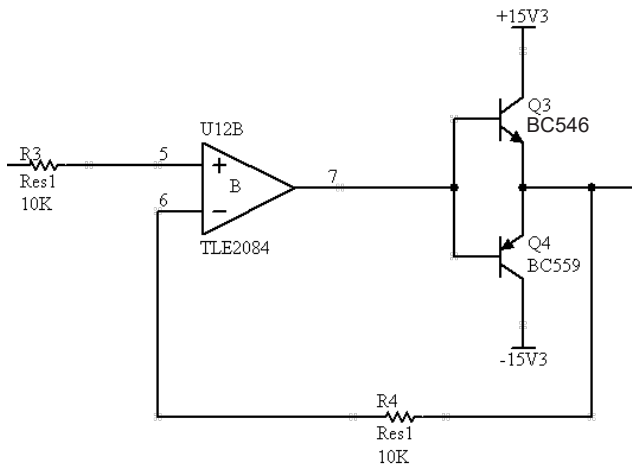
类似典型的 Cyclone 系列 FPGA 的 LVTTTL 电平驱动特性为 $\pm 25mA$ 。而当负载比较多，驱动电流比较大（例如几百 mA 等）时，都需要配置合理的驱动电路。可以考虑如下增加驱动的技术手段：

- 功率管驱动 例如 MOS 管
- 缓冲芯片驱动或 OC 门驱动芯片

例如 TI 或 FairChild 公司的 74ABT244、74ABT245 等等，它们的驱动电流可达： $-32mA I_{OH}$, $64mA I_{OL}$ 。

◆ 模拟量

例如常用 DA 卡的模拟输出为运放输出，其驱动能力非常有限，毫安级，如 PCI-2501 的驱动能力仅为 $5mA$ 。此时常采用三极管互补推挽驱动，以增大驱动电流。



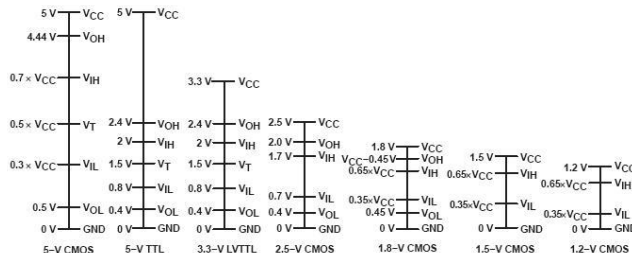
三极管互补推挽驱动电路原理简图

在图中，前级输入信号为 PCI-2501 的输出，驱动能力 $\pm 5mA$ ，为了增强驱动能力，增加了三极管驱动电路。由于信号是双极性的，因此采用互补的方式输出。为了消除互补输出三极管的死区，在两个互补的三极管之前又加了一个射级跟随器。

② 信号匹配

◆ 数字逻辑电平匹配

对于数字信号来说，随着技术的发展，电平种类越来越多也越来越复杂，如常见的 TTL，LVTTTL，CMOS 等等。那么这些信号之间由于工程的需求，往往需要互连。那么互连的一个原则就是“匹配”，而不能简单的进行物理连接。



常用逻辑电平标准示例简图

传感器与信号调理

◆ 模拟量输入量程 / 输出范围匹配

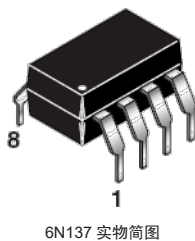
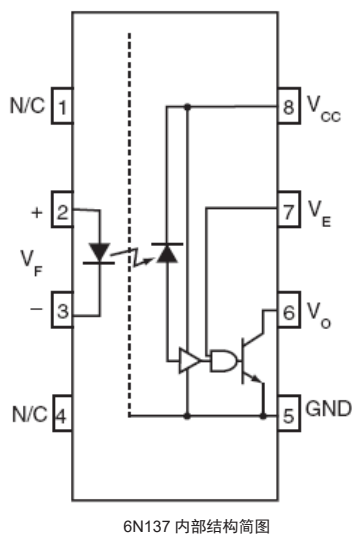
通用 AD 采集卡规定输入的信号和 DA 卡输出的模拟量信号均为所谓的“标准信号”，典型的如：0 ~ 5V、0 ~ 10V、±2.5V、±5V、±10V 等。但是用户的实际需求多种多样，例如某项目要求输出：0 ~ 15V、±15V、±30V 等“特殊”信号，还要求对 0 ~ 95V 等信号进行采集。这势必要求模拟量输入与输出之间的“匹配”。

数字信号调试技术

(1) 光耦隔离

光耦隔离，是一种把发光元件和光敏元件封装在同一壳体内，中间通过电 → 光 → 电的转换来传输电信号的半导体光电子器件。目前应用最广的是发光二极管和光敏三极管或光敏二极管组合成的光电耦合器。

例如常用的光耦隔离芯片 6N137，其内如结构如下图所示。



6N137 主要技术指标如下：

- ① 导通电流： $5\text{mA} < I_{th} < 15\text{mA}$
- ② 集电极开路输出
- ③ 最高速率：10MBit/s

使用光耦隔离芯片时应注意以下问题：

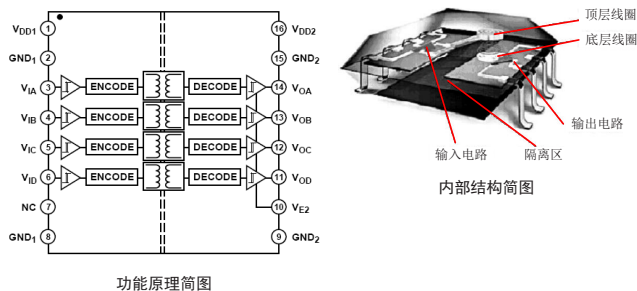
- ① 光耦隔离传输数字量时，要考虑光耦的响应速度；
例如：PC817 最高工作频率仅为 80KBit/s，6N137 最高工作频率可为 10MBit/s。
- ② 若输出有功率需求，还要考虑光耦的功率接口问题。
例如：6N137 的灌电流典型值为 13mA，PC817 的灌电流典型值为 30mA 左右，4N30 的灌电流典型值为 100mA 左右。如果驱动（控制）的负载电流更大，则光耦输出需要采用达林顿晶体管（ULN2803），或 MOS 管（VN0606）等，以进一步扩展输出电流。
- ③ 光耦隔离的缺陷：不稳定的电流传输率（CTR），非线性传输，受温度影响大和使用寿命有限。

(2) 非光耦隔离

非光耦隔离往往采用变压器作为隔离的手段。例如 ADuM140x 系列，其是 ADI (Analog device, inc) 公司推出基于其专利 iCoupler 磁耦隔离技术的通用型四通道数字隔离器。采用了高速 CMOS 工艺和芯片级的变压器技术，在性能、功耗、体积等各方面都有光电隔离器件无法比拟的优势。

ADuM140x 主要技术特点如下：

- ① 最高 90 Mbps 的数据传输速率；
- ② 最大 5.0 kV 额定隔离电压；
- ③ 与光电耦合器相比，PCB 电路板面积缩小 60%；
- ④ 每通道成本降低 40%，功耗降低 98%；
- ⑤ 直流校正功能，确保 DC 量准确传输。



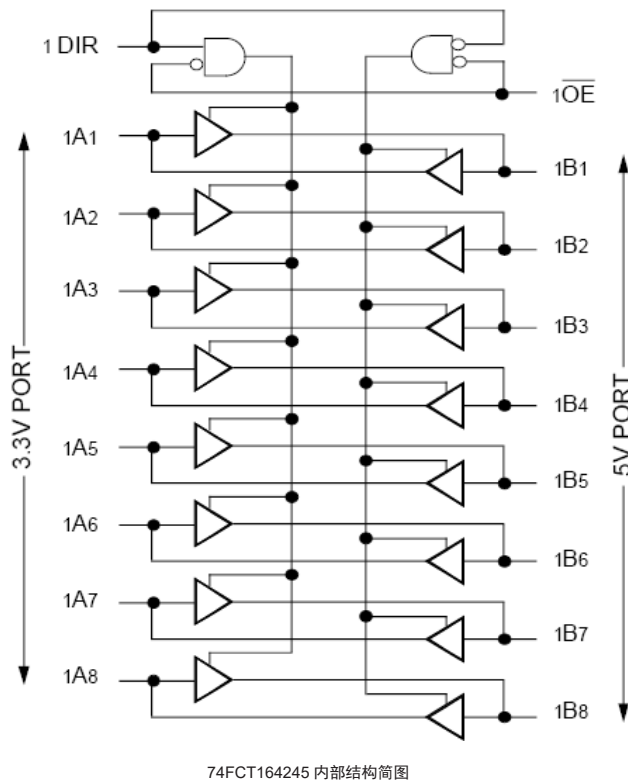
传感器与信号调理

(3) 逻辑电平匹配

逻辑电平匹配要遵循以下规则：

- ① 5V CMOS 可以驱动 5V TTL，反之则不可以；
- ② 3.3V LVTTTL 可以驱动 5V TTL，反之则不推荐。

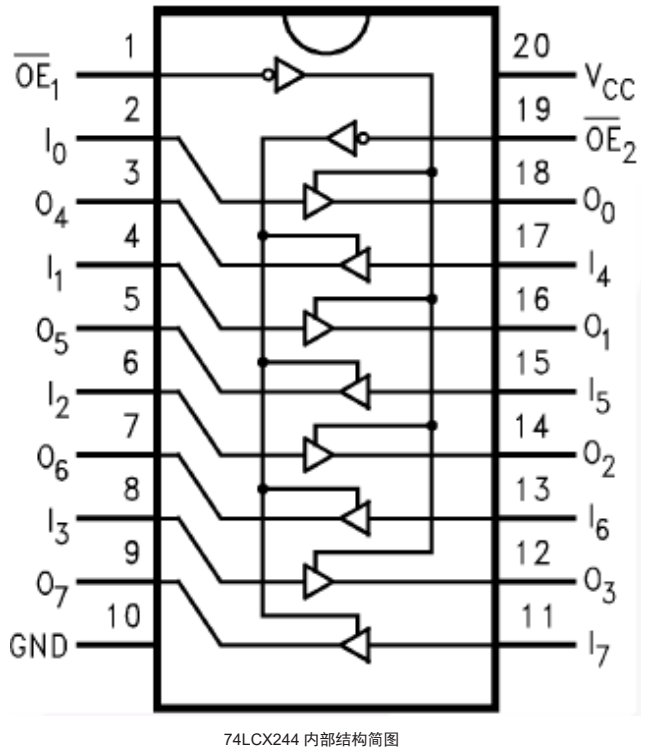
在进行逻辑电平匹配时，可采用专用的电平转换（匹配）芯片，如常用的 74FCT164245。



74FCT164245 主要技术特点如下：

- ① 采用 0.5 us CMOS 工艺
- ② 3.3V 和 5V 总线双向接口
- ③ 控制信号输入兼容 3.3V 和 5V TTL 电平
- ④ $VCC1 = 5V \pm 10\%$, $VCC2 = 2.7V$ to $3.6V$
- ⑤ 5V 总线侧输出驱动能力强, IOH 为 $-32mA$, IOL 为 $64mA$

另外常用的电平转换芯片还有 74LCX244, LCX244 包含八个具有三态输出非反相缓冲器。该设备可以用作存储器地址驱动器、时钟驱动器和面向总线的发送器 / 接收器。LCX244 是专为低电压 (2.5V 或 3.3V) 设计, 适用于连接到 5V 信号环境的 VCC 应用。



74LCX244 主要技术特点如下：

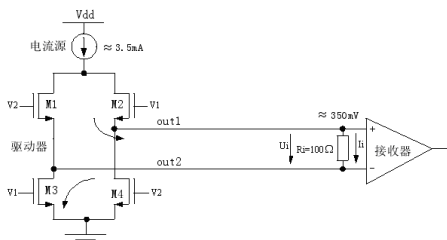
- ① 兼容 5V TTL 输入和输出
- ② VCC 电压范围从 2.3V 至 3.6V
- ③ 输出 ($VCC=3.0V$) 驱动能力 $\pm 24mA$

传感器与信号调理

(4) LVDS 信号调理

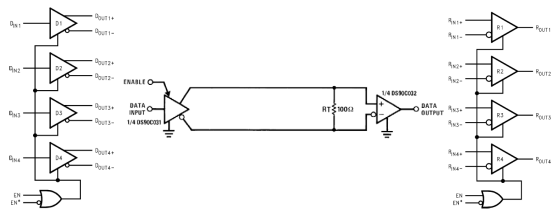
LVDS 即：Low Voltage Differential Signaling，是一种低电压摆幅的差分信号技术。其主要技术特点如下：

- ① 高速。相关标准建议 LVDS 的信号传输最大速率为 655Mbps
- ② 低功耗。CMOS 与电流环信号传输技术使得消耗的功率仅为 1.2mW
- ③ 低噪声。小输出电流 (3.5mA) 降低了电源 / 地噪声



LVDS 的工作原理示意图

LVDS 信号调理主要是 TTL/CMOS 电平与 LVDS 信号之间的互相转换。例如，某项目中自研开发的时间卡需要在 10m 范围内互相传送 10MHz 脉冲信号，方案将 TTL 信号转换为 LVDS 信号进行远距离传输，而后再反转换为 TTL 信号。项目方案中采用国家半导体的两个 LVDS 发送和接收芯片 DS90C032 和 DS90C031，实现与 TTL 信号之间的互相转换。这其中，DS90C032 为 4 通道 LVDS 接收器，DS90C031 为 4 通道 LVDS 发送器，二者接收和发送速率均超过 155Mbps。它们互连的原理如下图所示。



LVDS 调理工作原理示意图

模拟信号调试技术

(1) 模拟量隔离技术

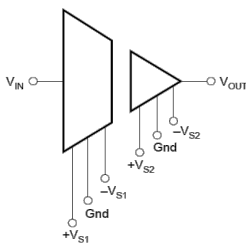
模拟量隔离常常采用隔离运算放大器，如 ISO124、AD210 等。隔离放大器可应用于高共模电压环境下的小信号测量，对被测对象和数据采集系统予以隔离，从而提高共模抑制比，同时保护电子仪器设备和人身安全。

① 使用隔离运放的目的：

- ◆ 隔离危险（高）电压
- ◆ 隔离危险（大）电流
- ◆ 隔离接地系统



隔离运放实物简图



隔离运放图形符号

② 隔离运放的分类：

隔离放大器按耦合方式的不同，可以分为变压器耦合，电容耦合和光电耦合三种。

a. 采用变压器耦合的隔离放大器主要包括：

- ◆ BURR—BROWN 公司（以下简称 BB 公司）的 ISO 212、3656

- ◆ Analog Devices 公司（以下简称 AD 公司）的 AD 202、AD204、AD210、AD215

b. 采用变压器耦合的隔离放大器主要包括：

- ◆ BB 公司的 ISO102、ISO103、ISO106、ISO107、ISO113、ISO120、ISO121、ISO122、ISO175

c. 采用光电耦合的隔离放大器主要包括：

- ◆ B 公司的 ISO 100、ISO 130、3650、3652

- ◆ 普公司（以下简称 HP）的 HCPL7800/7800A/7800B

传感器与信号调理

其中,有一些隔离放大器还提供了内置 DC/DC 变换器,以简化电路,节省空间,降低成本,提高性能,给使用者提供更大的灵活性。

此类隔离放大器包括:

◆ BB 公司的 ISO103、ISO107、ISO113、ISO212、ISO213、3656(三端隔离)

◆ AD 公司的 AD202、AD203、AD210(三端隔离)、AD215

下表列出了各种隔离放大器的基本参数。

型号	隔离电压	直流共模抑制比	60Hz 交流共模抑制比	增益	线性度	带宽	耦合方式	内置 DC/DC
ISO 100	705V	146dB	108dB	可变	0.07%	60kHz	光电	无
ISO 120	1500V	160dB	115dB	1V/V	0.01%	60kHz	光电	无
ISO 121	3500V	160dB	115dB	1V/V	0.01%	60kHz	电容	无
ISO 122	1500V	160dB	140dB	1V/V	0.02%	50kHz	电容	无
ISO 124	1500V		140dB	1V/V	0.005%	50kHz	电容	无
ISO 175	1500V	160dB	115dB	可变	0.052%	50kHz	电容	无
ISO 103	1500V	160dB	130dB	1V/V	0.025%	20kHz	电容	二端隔离
ISO 107	2500V	160dB	100dB	1V/V	0.025%	20kHz	电容	二端隔离
ISO113	1500V	160dB	130dB	1V/V	0.02%	20kHz	电容	二端隔离
3650/ 3652	2000V	140dB	120dB	可变	0.02%	15kHz	光电	无
3656	3500V	160dB	125dB	可变	0.05%	30kHz	变压器	三端隔离
AD 202	750V	130dB	110dB	1V/V	0.025%	2kHz	变压器	二端隔离
AD 204	750V	130dB	110dB	1V/V	0.025%	5kHz	变压器	二端隔离
AD 210	2500V	120dB	120dB	1V/V	0.012%	20kHz	变压器	三端隔离
AD 215	1500V	100dB	120dB	1V/V	0.005%	120kHz	变压器	二端隔离
HCPL 7800	3750V	76dB	/	8V/V	0.004%	100kHz	光电	无
HCPL 7850	/	69dB	/	8V/V	0.1%	100kHz	光电	无

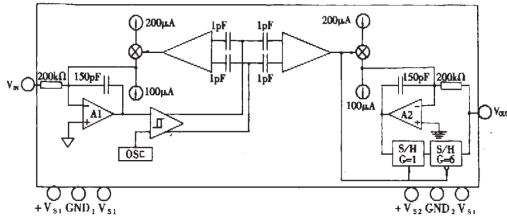
典型隔离放大器基本参数

传感器与信号调理

③ 隔离运放的工作原理

◆ ISO 122

ISO 122 隔离放大器的内部结构见下图。



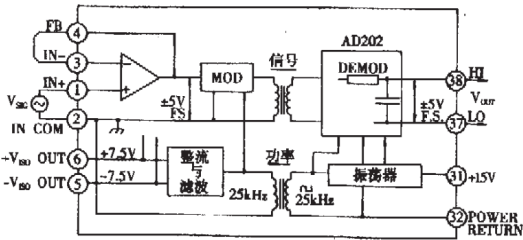
ISO 122 内部结构

在输入侧，输入信号 V_{IN} 经调制后，以数字量的形式由差分电容耦合到输出侧。调制过程通过切换 $\pm 100\mu A$ 电流源来实现。在输出侧，数字信号解调并滤除解调过程中产生的波纹，还原为模拟量信号输出。 V_{out} 经过 $20K\Omega$ 电阻反馈到加法器与 $\pm 100\mu A$ 电流源综合，从而使得 V_{out} 平均值正比于 V_{in} 。输出端的采样保持器则用于滤除解调过程中产生的固有波纹。

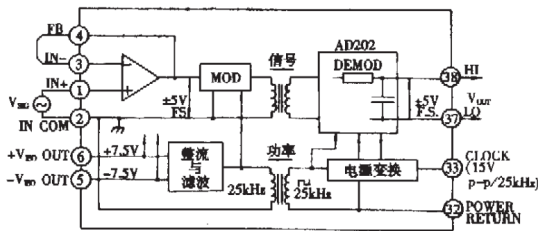
由于采用了数字化调制手段，隔离栅的性能不会影响到模拟信号的完整性，所以有较高的可靠性和良好的频率特性。ISO122 不需要外接其他元件就可以工作。其线性度优于 0.02% ，带宽 $50KHz$ 。输入侧和输出侧各需要一组 $\pm 15V$ 电源 ($\pm 4.5\sim\pm 18V$)。

◆ AD202/204

AD202/204 的内部结构见下列图。



AD 202 内部结构



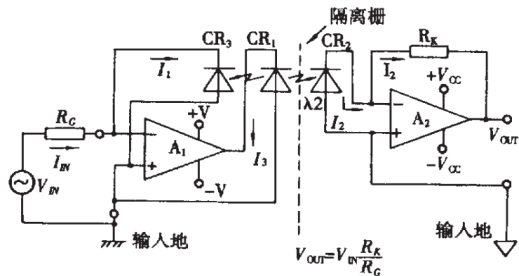
AD 204 内部结构

AD202/204 采用调幅手段，利用变压器将直流或交流信号耦合到输出侧。AD202/204 的输入端内置一个独立的运算放大器，可以用作输入的缓冲、提供必要的增益、或者用作滤波器、加法器、I/V 转换等等。对于有这类需要的电路来讲非常方便。AD202/204 内置一个 DC/DC 变换器，输出为 $\pm 7.5V$ 。可以提供电源给输入侧的运算放大器、调制器或其他电路。

AD202、AD204 的区别在于：AD202 直接由 $\pm 15V$ 电源驱动，而 AD204 由外部时钟提供电源。这样，在多通道情况下，几个 AD204 可以共用一个时钟源，降低了功耗和成本，提高了带宽，同时其内置 DC/DC 也可以提供更大的功率输出。

◆ 3650/3652

BB 公司的 3650、3652 为由线性光耦构成的隔离放大器。光电耦合器广泛用作数字电路中的隔离，若用于模拟电路，则有明显的缺点：一是非线性；二是稳定性受到环境温度和时间的的影响。需要采取特殊措施加以解决。3650、3652 的工作原理可以由下图说明。



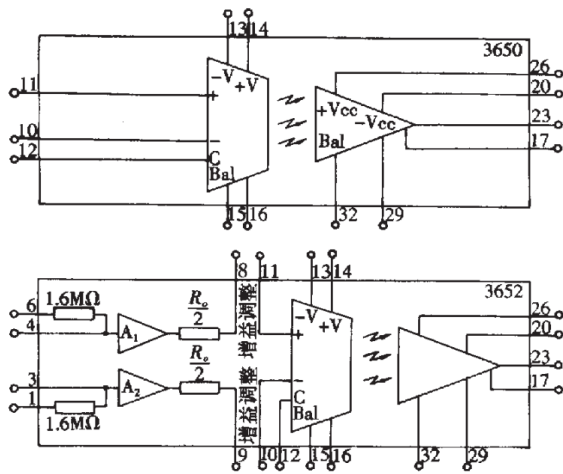
3650/3652 的工作原理

由放大器 A1，发光管 CR1 和光电管 CR3 构成负反馈回路。 $I_1=I_2=I_{IN}=V_{IN}/R_G$ 。光电管 CR2 和 CR3 是完全一致的，从 CR1 接收到的光量相同， $\lambda_1=\lambda_2$ ，则 $I_1=I_2=I_{IN}$ 。放大器 A2 与 R_k (内置电阻 $1M\Omega$) 构成 I/V 转换器， $V_{OUT}=12R_k I_2$ ，所以有： $V_{OUT}=V_{IN}R_k/R_G$ 。

这样，由于负责反馈回路的存在，解决了光电耦合器件非线性和不稳定的问题。只要 CR2 和 CR3 的一致性能得到保证，信号的耦合就不会受到光电器件性能的影响。

3650 和 3652 的区别在于前者没有输入缓冲放大器，见下图。

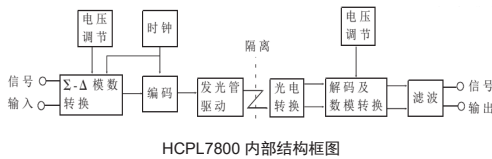
传感器与信号调理



3650 和 3652 的区别

◆ HCPL7800/7800A/7800B

惠普公司的 HCPL7800 是采用了 $\Sigma-\Delta$ AD 采用技术的光电隔离的另一种形式的隔离放大器。其结构框图见下图。



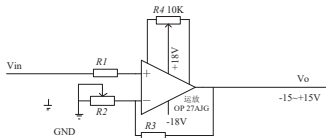
HCPL7800 内部结构框图

在隔离放大器输入侧，过采样模数转换器 $\Sigma-\Delta$ ADC 将输入的模拟信号转换为高速串行位流，其脉宽正比于信号的大小。位流经处理后由光电管以光的形式传送到输出侧，在输出侧解调，并由数模转换器重新转换为模拟量，经滤波后输出。

通过 $\Sigma-\Delta$ ADC 采样，将输入模拟量变换为位流，以数字脉冲的形式耦合到输出侧，光电器件的非线性和漂移问题得以解决，从而保证了隔离放大器的性能。

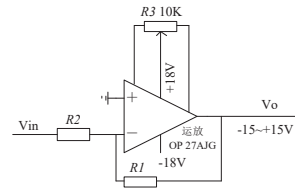
(2) 模拟量放大技术

模拟量放大常采用集成运算放大器构成的放大电路，典型的包括：同相放大电路和反相放大电路。典型同相放大电路如下图所示。



典型同相放大电路原理简图

典型反相放大电路如下图所示。

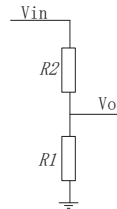


典型同相放大电路原理简图

(3) 模拟量衰减技术

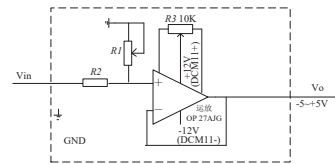
当被采集的电压幅值超过采集通道的量程范围时，就需要对电压信号进行衰减。其基本的原理都是采用电阻分压。单是简单的如下图所示采用两个电阻 R1 和 R2 进行分压，往往忽略了以下问题：

- ① 对前级电路的输入阻抗
- ② 对后级电路的输出阻抗
- ③ R1 和 R2 阻值的精度对分压比的影响



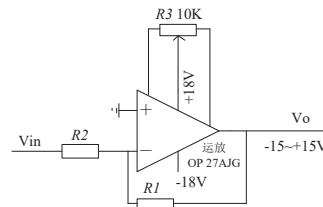
简单分压电路原理简图

为了对简单电阻分压电路的性能进行改善，往往在电阻分压电阻的后面加一级射级跟随电路，如下图所示。



典型射级跟随电路原理简图

另外，利用反相“放大电路”也可以构成衰减电路，如下图所示。

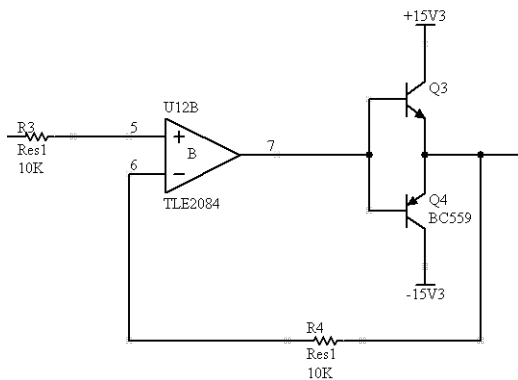


典型反相放大电路构成的衰减电路简图

传感器与信号调理

(4) 模拟量输出驱动技术

在测试系统或设备中，往往采用 PCI/CPCI/PXI 总线接口的 DA 卡作为模拟量的输出，但是这些模拟量输出通道本身的驱动能力有限，典型值约为 5mA。为了改善或提高其驱动负载的能力，常常采用三极管互补推挽驱动电路，增大驱动电流。其典型电路如下图所示。



典型互补推挽驱动电路简图

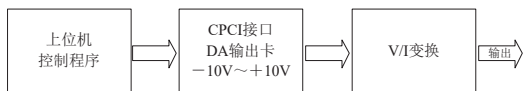
前级输入信号为 DA 卡的输出通道，驱动能力 $\pm 5\text{mA}$ ，为了增强驱动能力，增加了三极管驱动电路。由于信号是双极性的，因此采用互补的方式输出，为了消除互补输出三极管的死区，在两个互补的三极管之前又加了一个射级跟随器。

(5) V/I 变换技术

所谓 V/I 变换技术就是将电压量成比例的变换成电流量，即电压源变换成电流源。

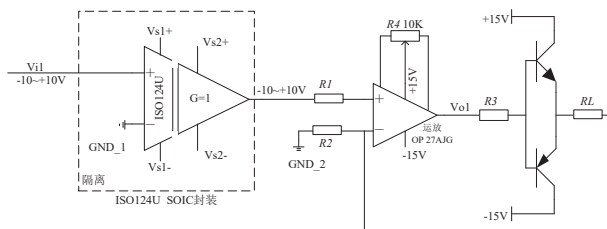
① 工程实例 I：某测试项目中技术指标要求 3 通道 $\pm 20\text{mA}$ 可编程调节恒流源输出，通道间隔离。

项目中 V/I 变换的总体方案如下图所示。



V/I 变换总体方案简图

3 个通道电流信号是通过 OP-27，1/1 放大，再经过隔离运放 ISO124P，将输入的地与输出的地完全隔离，现经过一个 OP-27 产生 $V_i/100\Omega$ 的电流，现经过两个类型相反的三极管产生正负电流。产品的输入阻抗要小于 500Ω 。

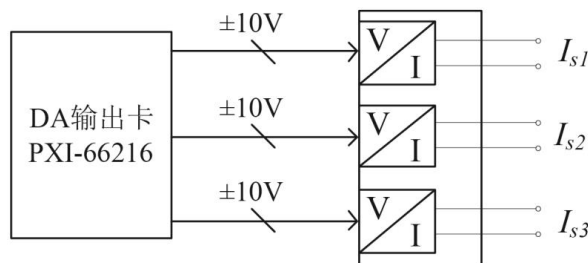


V/I 变换原理简图

② 实例 II：总控系统中要求 3 通道 $\pm 20\text{mA}$ 可编程调节恒流源输出，通道间隔离，电流的数字分辨率 $0.6\mu\text{A}$ ，稳态精度可达 $5\sim 6\mu\text{A}$ 。

项目方案综合考虑技术要求，并从系统集成性和紧凑性出发，采用高精度同步 DA 卡 + 高精度 V/I 变换调理的方案。

方案选用了一块简仪公司的 DA 卡 PXI-66216，它具有 16 路 16bits 模拟量输出通道，输出典型范围 $\pm 5\text{V}$ 。选取其中的三路电压送至主控调理 III，由其实现高精度 V/I 成比例变换调理，最终向产品输出独立的三路电流注入信号。电流的理论数字分辨率为 $0.6\mu\text{A}$ ，稳态精度可达 $5\sim 6\mu\text{A}$ 。同时该卡还有多达 32 路的 AD 通道，可以方便的实现电流注入的自检与校准功能。



V/I 变换调理

电流注入总体方案简图

V/I 变换调理的主要功能是实现高精度 V/I 成比例变换调理，最终向产品输出独立的三路电流注入信号。V/I 变换调理的原理如下图所示。其包括以下组成部分：

◆ 模拟电压隔离部分

以隔离运放 ISO124 为核心构建，构成三路模拟电压隔离调理电路。同时，该部分在 ISO124 的后端还设计有以 OP27 为核心构成的 ≈ 1 倍正向放大电路。该部分将输入的 3 路共地的 $+10\text{V}\sim -10\text{V}$ 范围内变化的控制电压信号经 ISO124 线性变换至 $+10\text{V}\sim -10\text{V}$ 范围，以便充分利用 ISO124 的线性度特性。正向放大电路可以对 ISO124 输出的电压信号线性度和零点进行调节。

传感器与信号调理

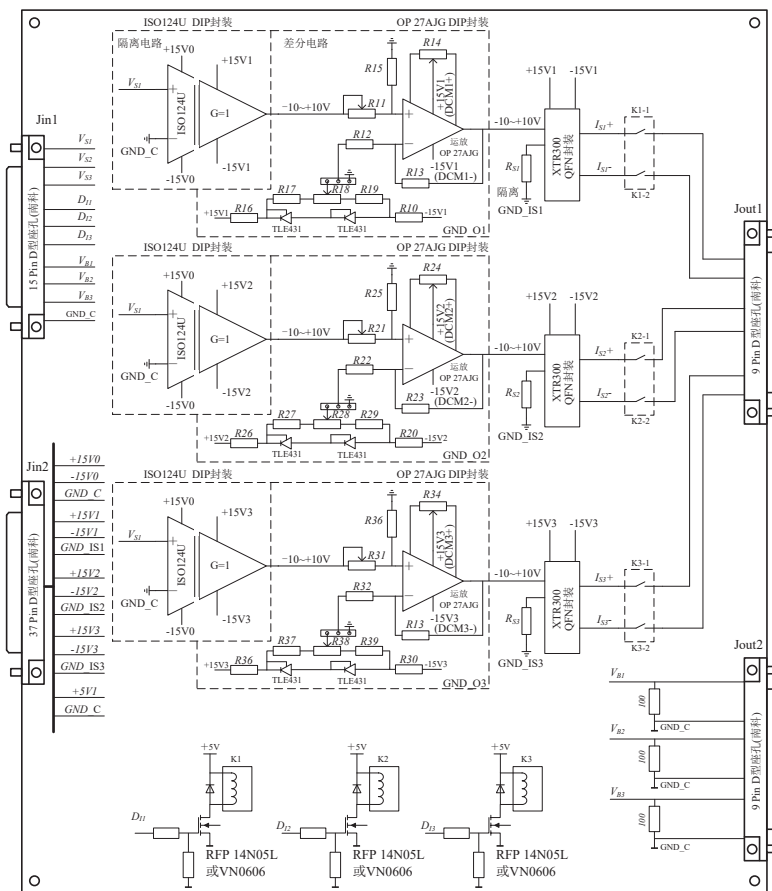
◆ V/I 变换部分

以 Burr-Brown 公司的高精度 V/I 变换芯片 XTR300 为核心构建。一个通道利用一片 XTR300。XTR300 可将输入电压呈比例线性变换为电流信号。电流输出最大可达 $\pm 24\text{mA}$ 。线性度精度为 $\pm 0.01\% \text{FS}$ ，线性度温度漂移 $\pm 1.5\text{ppm}/^\circ\text{C}$ ，增益精度为 $\pm 0.04\% \text{FS}$ ，实测电流摆率为 $4\text{mA}/\mu\text{s}$ 。

同时，在 XTR300 V/I 变换电路部分的前级还有差分一个差分电路，其主要作用是调节零点和增益。

◆ 电流注入自检和校准部分

将输出的电流通过 DB9 芯插座引回板卡，每路电流通道串接 100Ω 高精电阻，将电流量变换为电压量后再输出，以便 AD 采集通道进行采集。



V/I 变换调理原理简图

上述方案的特点如下：

- PXI 总线接口
- 3 路电流同步输出
- 稳态精度高：精度可达 $4\sim 5\mu\text{A}$
- 分辨率高：16bits
- 动态特性好：XTR300 电流摆率可达 $4\text{mA}/\mu\text{s}$
- 集成自检功能

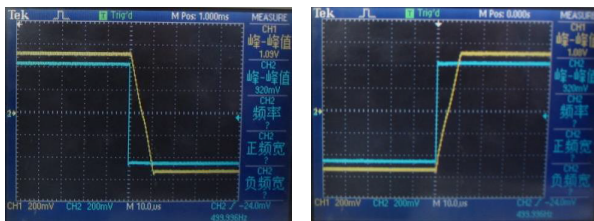
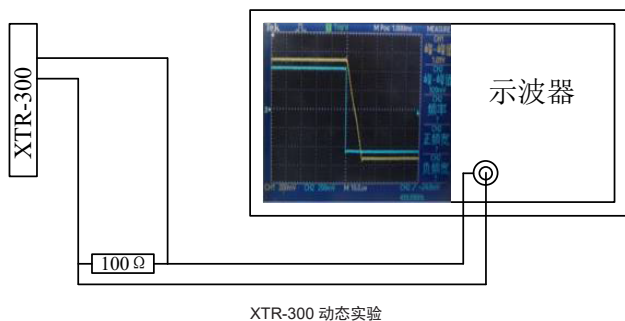
其中，简仪公司的同步 DA 输出卡 PXI-66216 具有很好的精度，其最大量程 ($\pm 10\text{V}$) 输出的精度可达 2mV ，也就是 0.02% ，其输出电压建立时间为 $2\mu\text{s}$ 左右。XTR300 的线性度精度 ($\pm 0.01\% \text{FS}$) 和增益精度 ($\pm 0.04\% \text{FS}$) 较高，其是 V/I 变换的核心组成部分。实验对 XTR300 也进行了相关测试。

传感器与信号调理

XTR300 精度测试			
输入电压 (V)	实际输出电流 (mA)	理论输出电流 (mA)	电流偏差 (uA) (实际-理论)
-3.9915	-18.267	-18.270	3
-3.6461	-16.685	-16.689	4
-3.3782	-15.457	-15.462	5
-3.0197	-13.818	-13.822	4
-2.9291	-13.401	-13.407	6
-2.5764	-11.787	-11.792	5
-2.2419	-10.256	-10.261	5
-2.0477	-9.368	-9.373	5
-1.8715	-8.561	-8.566	5
-1.6276	-7.444	-7.450	6
-1.3772	-6.300	-6.304	4
-1.0300	-4.7098	-4.7146	4.8
-0.9278	-4.2427	-4.2468	4.1
-0.7209	-3.2962	-3.2998	3.6
-0.5271	-2.4095	-2.4127	3.2
0.5533	2.5337	2.5326	1.1
0.7807	3.5744	3.5734	1
0.9265	4.2418	4.2409	0.9
1.2554	5.744	5.746	-2
1.4504	6.636	6.639	-3
1.6282	7.449	7.453	-4
1.8834	8.618	8.621	-3
2.0414	9.340	9.344	-4
2.3814	10.898	10.900	-2
2.5762	11.790	11.792	-2
2.8819	13.189	13.191	-2
3.0318	13.877	13.877	0
3.2643	14.943	14.942	1
3.6273	16.608	16.603	5
3.8645	17.697	17.699	-2

传感器与信号调理

• XTR300 动态测试



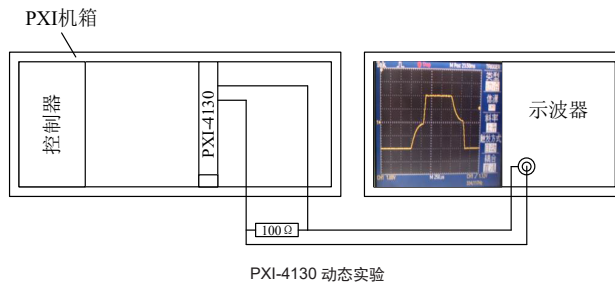
XTR-300 动态实验结果简图

从上述动态实现发现，XTR300 电流摆率可达 $4\text{mA}/\mu\text{s}$ 。

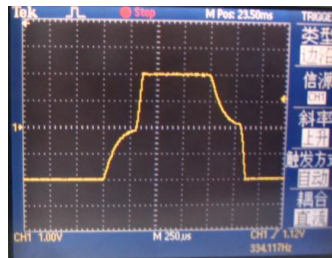
另外，我们也注意到 PXI 总线接口电流输出卡 PXI-4130。显然，如采用 PXI-4130 势必选用其 $\pm 20\text{mA}$ 量程。但是实际静态实验发现，PXI-4130 在 $\pm 20\text{mA}$ 量程时，不确定度为 $\pm 4\mu\text{A}$ 。即 $\pm 4\mu\text{A}$ 范围内电流无法精确控制。而 $\pm 4\mu\text{A}$ 范围以外，其静态的编程分辨率为 $1\mu\text{A}$ 。PXI-4130 的静态精度实验结果如下图所示。

PXI-4130 静态精度实验简表		
编号	设定值 (mA)	实测值 (mA)
1	20	20.008
2	18	18.009
3	16	16.007
4	14	14.006
5	12	12.004
6	10	10.005
7	8	8.004
8	6	6.004
9	4	4.001
10	2	2.001
11	0.4	0.399
12	-0.4	-0.399
13	-2	-2.002
14	-4	-4.002
15	-6	-6.005
16	-8	-8.006
17	-10	-10.007
18	-12	-12.008
19	-14	-14.007
20	-16	-16.009
21	-18	-18.009
22	-20	-20.009

同时，又进行了动态实验（如下图所示）。利用上层 Windows 的软件程序发送一定频率的（典型频率 400Hz ）方波电流，通过示波器观测其电流的上升沿与下降沿，研究其电流动态变化特性。



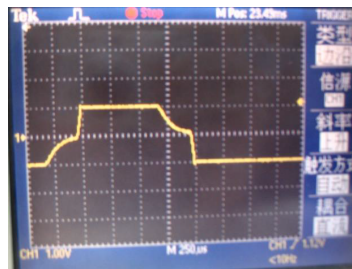
• 400Hz 方波电流，50% 的 $+20\text{mA}$ 电流，50% 的 -20mA 电流



动态电流变化 I 波形

从图中可以看出，电流上升波形和下降波形非常相似，由两段构成，一段类似指数函数，另一段呈线性。上升和下降时间基本一样，约为 $350\mu\text{s}$ 。

• 400Hz 方波电流，50% 的 $+10\text{mA}$ 电流，50% 的 -10mA 电流

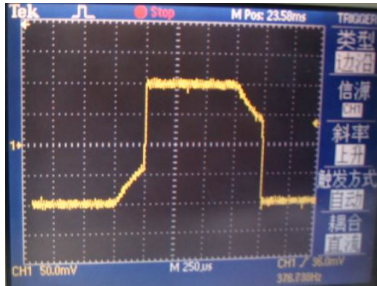


动态电流变化 II 波形

从图中可以看出，电流上升波形和下降波形时间约为 $350\mu\text{s}$ 。

传感器与信号调理

• 400Hz 方波电流，50% 的 +1mA 电流，50% 的 -1mA 电流



动态电流变化 III 波形

从图中可以看出，电流上升波形和下降波形时间约为 250µs。总之，只要存在 mA 级的电流正负交越变换，其动态建立过程均非常相似，电流建立时间约为 350µs。只有到小于 1mA 的电流正负交越变换，电流建立时间才能有所减小。综合考虑实际电流注入的应用，可以认为其建立时间约为 350µs。

电源调理与控制技术

测试设备中通常要求对直流 (DC) 供电电压和供电电流进行监控。有以下几种解决方案：

(1) 高档的 DC 直流电源：可通过 GPIB 接口或 RS232 等接口对电源的通 / 断进行控制，读取电压电流。

例如：LAMBDA 可调直流稳压电源 GEN60-55，具有 GPIB 接口。



LAMBDA 电源实物简图

固纬可调开关直流电源，具有 RS232 接口。



固纬可调电源实物简图

Agilent 可调直流电源 E3634，具有 GPIB 接口和 RS232 接口。



Agilent 电源实物简图

(2) 常规的 DC 直流电源：无 GPIB 接口或 RS232 等接口。

解决方案——外加电源调理，主要实现以下功能：电源通断控制、供电电压 / 供电电流变换采集、数字显示等。

① 通断控制

机械继电器：导通电阻小，基本无压降，响应频率低，延时较大。



OMRON 单极机械继电器

其主要技术特性如下表所示。

种类 负载	1a		2a	
	电阻负载 (cosφ=1)	感性负载 (cosφ=0.4)	电阻负载 (cosφ=1)	感性负载 (cosφ=0.4)
额定负载	AC250V 15A DC 24V 15A	AC250V 10A DC 24V 7.5A	AC250V 10A DC 24V 10A	AC250V 7.5A DC 24V 7.5A
额定通电流	15A		10A	
接点电压的最大值	AC250V、DC125V			
接点电流的最大值	15A		10A	
开关容量的最大值 (参考值)	3750VA 375W	2500VA 255W	2500VA 245W	1850VA 120W

OMRON 单极机械继电器技术特性表

固态继电器：导通管压降大，压降非线性，响应频率高，延时较小。固态继电器 (Solid State Relay, 缩写 SSR)，固态继电器有三部分组成：输入电路，隔离 (耦合) 和输出电路。

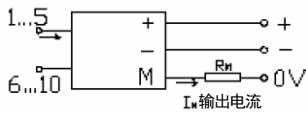
传感器与信号调理



固态继电器实物简图

② 供电电流采集

- ◆ 电流霍尔传感器 适合电流较大场合
- ◆ 采样电阻 适合电流较小场合



宇波模块接线图



宇波模块外形简图

宇波模块性能参数如下表所示。

项目	型号	CHB-25NP				
I_N	额定电流 (RMS)	5A	6A	8A	12A	25A
I_P	测量范围	0~±7A	0~±9A	0~±12A	0~±18A	0~±36A
I_M	测量电流 (输出电流)	25mA	24mA	24mA	24mA	25mA
K_N	匝数比	5:1000	4:1000	3:1000	2:1000	1:1000
R_M	测量电阻	R_{Mmin}		R_{Mmax}		
	$V_C = \pm 15V$	100Ω (25A 或 36A 时)		300Ω (25A 时); 200Ω (36A 时)		
X	精度	I_N 的±0.8%				
V_C	电源电压	±15V (±5%)				

宇波模块性能参数简表

总线接口技术

引言

接口总线是测控系统中各种信息的传送通路，它将计算机和各种设备仪器连接起来组成测控系统，是测控系统的重要组成部分。因此，在设计以计算机为中心的测控系统时，设计和选择合适的接口成为系统设计的重要环节。接口和总线是相辅相成的两个方面，总线更多的是指一种规范、一种结构形式，其主要特征是共享传输介质，而接口多指完成通信的硬件系统，是外部设备与计算机连接的端口。

(1) 总线的概念

总线是一组信号线的集合，是连接有若干设备并进行信号传送的一条信号线或者一组信号线，是实现将信号从任意一个源点到任意一个终点的一组传送数字信号的公共通道。它的特点是共用性，即可同时挂接多个模块或设备，可在模块与模块之间、设备与设备之间传送信息。它的核心功能是传输。

总线包括三个部分：

数据总线 DB (Data Bus) 用来传送数据，其位数一般与微处理器字长相同。数据总线具有双向功能。通过它可以实现 CPU、存储器和输入输出接口之间的数据交换。

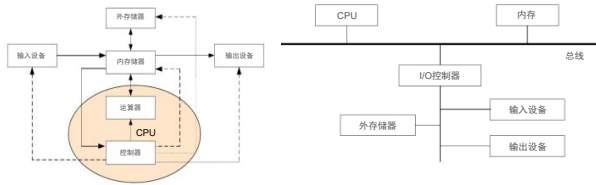
地址总线 AB (Address Bus) 用来传送地址信息。它是单向传送的，用来把地址信息从 CPU 单向地传送到存储器或 I/O 接口，指出相应的存储单元或 I/O 设备。16 位 AB 能直接寻址的存储空间为 2¹⁶，存储地址编址范围为 0000H~FFFFH。

控制总线 CB (Control Bus) 用来传输控制信号。这些控制信号控制着计算机按一定的节拍，有规律地自动工作。控制总线的多少因不同性能的 CPU 而异。

总线的类别很多，按其传输数据的方式可分为串行总线和并行总线；按应用的场合可分为芯片总线、板内总线、机箱总线、设备互联总线、现场总线和网络总线等；按用途可分为计算机总线、外设总线和测控系统总线；按总线的作用域可分为全局总线和本地总线，按层次结构可分为内部总线、系统总线、外部总线，等等。本章主要针对串行总线和并行总线的这两种类别对总线接口技术进行说明。

(2) 采用总线接口技术的优点及必要性

任何一种微处理器都要与一定数量的外部存储器和外围设备连接，再配以必要的软件，即构成微型计算机系统，它才能工作，才能发挥其巨大的作用。但是如果将外部存储器和每一台外围设备都用一组线路与 CPU 连接，那么线路会错综复杂，甚至难以实现。为了简化硬件电路的设计，简化系统结构，通常只用一组线路，配置以适当的接口电路，实现 CPU 与存储器以及各外围设备的连接，即采用总线技术。



计算机、测控系统等采用总线结构设计以后，体现出了很多优越性，这些优越性概括起来如下面几点：

- ◆ 简化系统设计
- ◆ 简化系统结构，提高系统可靠性
- ◆ 便于系统的扩充和更新
- ◆ 便于维修
- ◆ 经济性好
- ◆ 能得到多家厂商的支持
- ◆ 便于组织生产

串行总线

串行通信的基本特征是数据逐位顺序进行传送，串行通信的格式及约定（如同步方式、通讯速率、数据块格式、信号电平等）不同，形成了多种串行通信的协议与接口标准。

串行传输及串行通信，串行传输是指数据的二进制代码在一条物理信道上以位为单位按时间顺序逐位传输的方式。串行传输时，发送端逐位发送，接收端逐位接受，同时，还要对所接受的字符进行确认，所以收发双方要采取同步措施。

总线接口技术

串行传输有以下几个特点：

- ◆ 串行传输在一根传输线上既传送数据信息又传送联络控制信号。
- ◆ 有固定的数据格式，分异步与同步数据格式。
- ◆ 串行通信中对信号的逻辑定义与 TTL 不兼容，故需要进行逻辑关系与逻辑电平转换。
- ◆ 串行传送信息的速率需要控制，要求双方设定通信传输的波特率。
- ◆ 传输线路简单，利用简单的线缆就可实现通信，降低成本，适用于远距离通信，但传输速度慢的应用场合。

(1) 异步通信和同步通信

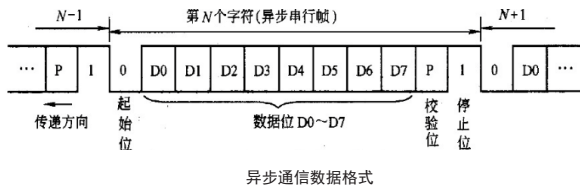
串行通信按同步方式可分为异步通信和同步通信。

① 异步通信

异步通信依靠起始位、停止位保持通信同步。异步通信数据按帧传输，一帧数据包含起始位、数据位、校验位、和停止位。数据以字符为单位传输。当发送一个字符代码时，自发前面要加一个“起始”信号，其长度为一位，极性为“0”即空码 (Space) 状态；规定在线路不传送数据时全部为“1”即传号 (Mark) 状态。字符后边要加一个“停止”信号，其长度为 1, 1.5 或 2 位，极性为“1”。自负本身的长度为 5~8 位数据，视传输的数据格式而定。例如，当传送的数字（或字符）用 ASCII 码表示时，其长度为七位。在某些传输中，为了减少误码率，经常在数据之后还加一位“校验位”。

由此可见，一个字符由起始位 (0) 开始，到停止位 (1) 结束，其长度为 7~12 位。起始位和停止位用来区分字符。传送时，字符可以连续发送，也可以断续发送。不发送字符时线路保持“1”状态。字符发送的顺序为先地位后高位。

异步通信对硬件要求低，实现起来比较简单、灵活，适用于数据的随机发送/接收，但因每个字节都要建立一次同步，即每个字符都要额外附加两位，所以工作速度较低，在单片机中主要采用异步通信的方式。

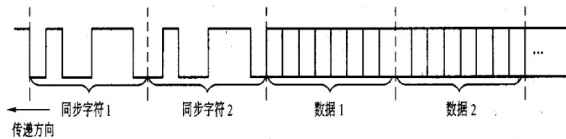


异步通信数据格式

② 同步通信

同步通信是依靠同步字符保持通信同步。同步通信是由 1~2 个同步字符和多字节数据位组成，同步字符作为起始位以触发同步时钟开始发送或接收数据；多字节数据之间不允许有空隙，每位占用的时间相等；空闲位需发送同步字符。同步通信的特点不仅是字符内部保持同步，而且字符与字符之间也是同步的。在这种通信方式下，收/发双方必须建立准确的位定时信号，也就是说收/发时钟的频率必须严格地一致。同步通信在数据格式上也与异步通信不同，每个字符不增加任何附加位，而是连续发送。但是在传送中，数据要分成组（帧），一组含多个字符代码或若干个独立的码元。

为使收/发双方建立和保持同步，在每组的开始处加上规定的码元序列，作为标志序列。在发送数据之前，必须先发送此标志序列，接收端通过检测该标志序列实现同步。



同步通信数据格式

(2) 串行通信的数据传送方式

在串行通信中，数据传送有三种方式：单工方式、半双工方式和全双工方式。

① 单工方式 (Simplex Mode)

在这种方式中，只允许数据按一个固定的方向传送，如图 (a) 所示。图中 A 只能发送数据，称为发送器 (Transfer); B 只能接收数据，叫做接收器 (Receiver)。而数据不能从 B 向 A 传送。

② 半双工方式 (Half-Duplex Mode)

半双工方式如图 (b) 所示。在这种方式下，数据既可

总线接口技术

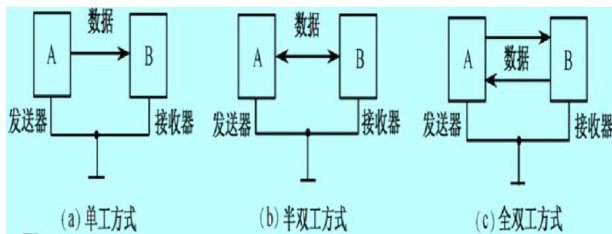
以从 A 传向 B，也可以从 B 向 A 传输。因此，A，B 既可以作为发送器，又可作为接收器，通常称为收发器 (Transceiver)。从这个意义上讲，这种方式似乎为双向工作方式。但是，由于 A，B 之间只有一根传输线，所以信号只能分时传送。

即在同一时刻，只能进行一个方向传送，不能双向同时传输。因此，将其称为“半双工”方式。在这种工作方式下，要么 A 发送，B 接收；要么 B 发送，A 接收。当不工作时，令 A，B 均处于接收方式，以便随时响应对方的呼叫。

③ 全双工方式 (Full-Duplex Mode)

虽然半双工方式比单工方式灵活，但它的效率依然比较低。主要原因是从发送方式切换到接受方式需要一定的时间，大约为数毫秒。重复线路切换所引起的延迟积累时间是相当可观的。另一方面，也是更重要的，就是同一时刻只能工作在某一种方式下，这是半双工效率不高的根本原因所在。

解决的方法是增加一条线，使 A，B 两端均可同时工作在收发方式，如图 (c) 所示，将图 (c) 与图 (b) 相比，虽然对每个站来讲，都有发送器和接收器，但由于图 (c) 中有两条传输线，用不着收发切换，因而传送速率可成倍增长。



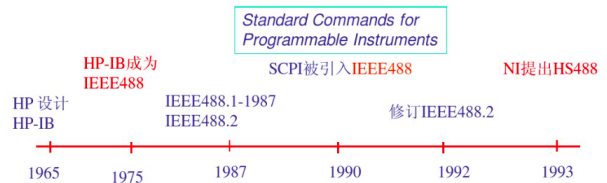
值得说明的是，全双工与半双工方式比较，虽然信号传送速度大增，但它的线路也要增加一条，因此系统成本将增加。在实际应用中，特别是在异步通信中，大多数情况都采用半双工方式。这样，虽然发送效率较低，但线路简单、使用，对于一般系统也能胜任。

(3) GPIB 总线

① GPIB 总线概述

GPIB (General-Purpose Interface Bus) - 通用接口总线，大多数台式仪器是通过 GPIB 线以及 GPIB 接口与电脑相连。1965 年，惠普公司 (Hewlett-Packard) 设

计了惠普接口总线 (HP-IB, 用于连接惠普的计算机和可编程仪器)。由于其高转换速率 (通常可达 1Mbytes/s)，这种接口总线得到普遍认可，并被接收为 IEEE 标准 488-1975 和 ANSI/IEEE 标准 488.1-1987。后来，GPIB 比 HP-IB 的名称用得更广泛。ANSI /IEEE 488.2 -1987 加强了原来的标准，精确定义了控制器和仪器的通讯方式。可编程仪器的标准命令 (Standard Commands for Programmable Instruments, SCPI) 采纳了 IEEE488.2 定义的命令结构，创建了一整套编程命令。GPIB 的发展状况如下图：



② GPIB 总线的特性

◆ 器件容量：可以用一条总线互相连接若干个器件，以组成一个自动测试系统。包括计算机在内，系统中可连接的器件数目最多不超过 15 个。这主要受目前 TTL 接口驱动器最大驱动电流的影响。当组建测试系统所需要的器件数目多余 15 个时，只需在计算机内再添加一个 IEEE 488 接口。通过这个接口可以再连接 14 个器件。

◆ 传输距离：整个测试系统的连接电缆的最大长度不超过 20m。

◆ GPIB 总线是一种异步双向型总线，数据传输采用并行比特 (bit)、串行字节 (B) 双向异步传送方式，其最大传输速率不超过 1MB / s。

◆ 总线上传输的消息采用负逻辑，低电平 ($\leq +0.8V$) 为逻辑“1”，高电平 ($\geq +2.0V$) 为逻辑“0”。

◆ 地址容量：系统中每一个仪器都设置有一个地址，IEEE 488 协议规定地址为 5 位，可得到 $2^5=32$ 个地址。

◆ 工作方式：在一个 GPIB 标准接口总线系统中，要进行有效的通信联络至少有“讲者”、“听者”、“控者”三类仪器装置。

“讲者”是通过总线发送仪器消息的仪器装置 (如测量仪器、数据采集器、计算机等)，在一个 GPIB 系统中，可以设置多个讲者，但在某一时刻，只能有一个讲者

总线接口技术

在起作用。

“听者”是通过总线接受由讲者发出消息的装置（如打印机、信号源等），在一个 GPIB 系统中，可以设置多个听者，并且允许多个听者同时工作。

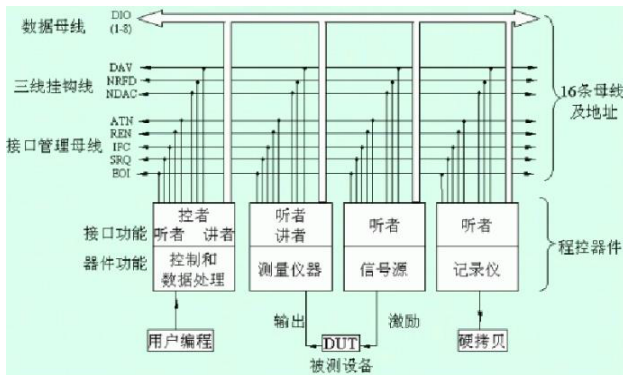
“控者”是数据传输过程中的组织者和控制者，例如对其他设备进行寻址或允许“讲者”使用总线等。控者通常由计算机担任，GPIB 系统不允许有两个或两个以上的控者同时起作用。

控者、讲者、听者、被称为系统功能的三要素，对于系统中的某一装置可以具有三要素中的一个、两个或全部。GPIB 系统中的计算机一般同时兼有三者的功能。

◆ 一般适用于电气干扰轻微的实验室和生产现场。

③ GPIB 总线的信号定义及数据传输机制

GPIB 总线结构如下图所示：



GPIB 总线结构

GPIB 总线是一条 24 芯的无源电缆线，包括 16 根信号线和 8 根地线，其中有 8 根双向 8 位数据线、3 根握手线、5 根接口管理控制线。GPIB 总线上的信息按位并行、字节串行的方式传送。

a. 信号定义

• 数据线 (DIO0 ~ DIO7)

作用：传递仪器消息和大部分接口消息，包括数据、命令和地址。由于这一标准没有专门的地址总线和控制总线，因此必须用其余两组信号线来区分数据总线上的信息类型。

• 握手线 (DAV, NRFD 和 NDAC)

作用：控制数据总线的时序，以保证数据总线能正确、有节奏的传输信息，这种传输技术称为三线挂钩技术。3 根握手线的定义如下：

DAV(Date Valid): 数据有效信号线。当 $DAV = 1$ (低电平) 时。DIO 线上的数据有效。

NRFD(Not Ready for Data): 未准备好接受数据信号线。当 $NRFD = 1$ 时，表示至少有一个设备未准备好接收数据；当各接收设备都准备好接收数据是， $NRFD = 0$ (高电平)。

NDAC(Not Data Accepted): 未接收到数据信号线，当 $NDAC = 1$ 时，表示至少有一个设备未接收到数据。

• 接口管理控制线 (ATN, IFC, REN, EOI 和 SRQ)

作用：控制 GPIB 总线接口的状态。5 根控制线的定义如下：

ATN: 引起注意信号。当 $ATN = 1$ 时，表明总线上所载的信息为当前控者向各个设备发送的总线命令或者地址；当 $ATN = 0$ 时，表明总线上所载信息为讲者向已寻址的听者发送的程控命令、数据字节或者状态字节。

IFC: 接口清零。它使系统处于初始状态 (系统复位)。

SRQ: 服务请求。当 $SRQ = 1$ 时表明总线上有一个器件需要引起控者注意 (类似于中断请求信号)，这时控者会查询所有器件，看是哪个器件发出的中断请求。

REN: 远程启动。它使设备的方式码与其它代码一起置位，以便远距离操作或本地操作。 $REN = 1$ 为工作于远程控制的先决条件。

EOI: 结束标识。它告诉控制装置数据传输已经结束。

b. 三线挂钩通信联络过程

在 GPIB 系统中，每传递一个字节的数字信息，源方 (讲者与控者) 与受方 (听者) 之间都要进行一次三线挂钩过程。

假设地址已发送，听者和讲者均已受命。三线挂钩过程如下：

• 听者使 NRFD 呈高电平，表示已做好接收准备，总

总线接口技术

线上所有听者是“线或”连接至 NRFD 线上，因此只要有一个听者未做好准备，NRFD 就呈低电平。

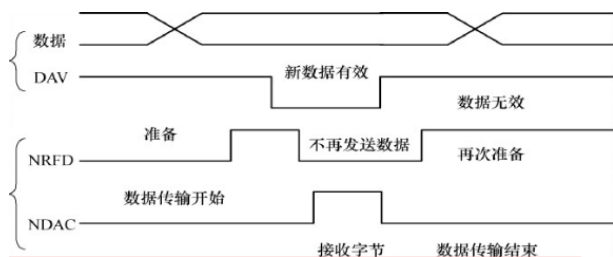
- 讲者发现 NRFD 呈高电平后，就把数据放在 DIO 线上，并令 DAV 为低电平，表示 DIO 线上的数据已经稳定且有效。

- 听者发现 DAV 线呈低电平，就令 NRFD 呈低电平，表示准备接收数据。

- 在接收数据的过程中，NDAC 线一直保持低电平，直至每个听者都接收完数据，才上升为高电平。所有听者也是“线或”接到 NDAC 线上。

- 当讲者检出 NDAC 为高，就令 DAV 为高，表示总线上的数据不再有效。

- 听者检出 DAV 为高电平，就令 NDAC 再次变为低电平，以准备进行下一个循环过程。



三线挂钩通信联络过程

显然，三线挂钩技术可以协调快慢不同的设备可靠地在总线上进行信息传递。

(4) RS-232\422\485 总线

① RS-232 的历史和作用

在串行通讯时，要求通讯双方都采用一个标准接口，使不同的设备可以方便地连接起来进行通讯。RS-232-C 接口（又称 EIA-RS-232-C）是目前最常用的一种串行通讯接口。（“RS-232-C”中的“-C”只不过表示 RS-232 的版本，所以与“RS-232”简称是一样的）

它是在 1970 年由美国电子工业协会 (EIA) 联合贝尔系统、调制解调器厂家及计算机终端生产厂家共同制定的用于串行通讯的标准。它的全名是“数据终端设备 (DTE) 和数据通讯设备 (DCE) 之间串行二进制数据交换接口技术标准”该标准规定采用一个 25 个脚

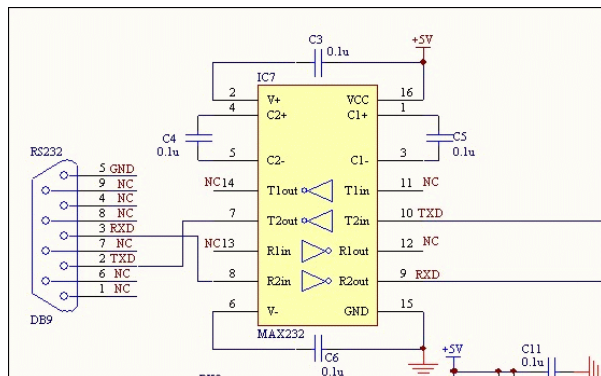
的 DB-25 连接器，对连接器的每个引脚的信号内容加以规定，还对各种信号的电平加以规定。后来 IBM 的 PC 机将 RS232 简化成了 DB-9 连接器，从而成为事实标准。而工业控制的 RS-232 口一般只使用 RXD、TXD、GND 三条线。

② RS-232 串口的电气特性：

- ◆ RS-232 串口通信最远距离是 50 英尺。

- ◆ RS232 可做到双向传输，全双工通讯，最高传输速率 20kbps。

- ◆ RS-232C 上传送的数字量采用负逻辑，且与地对称。逻辑 1: -3 ~ -15V。逻辑 0: +3 ~ +15V。所以与单片机连接时常常需要加入电平转换芯片：



③ 串口通信参数：

- ◆ 波特率：RS-232-C 标准规定的数据传输速率为每秒 50、75、100、150、300、600、1200、2400、4800、9600、19200 波特。

- ◆ 数据位：标准的值是 5、7 和 8 位，如何设置取决于你想传送的信息。比如，标准的 ASCII 码是 0 ~ 127(7 位)；扩展的 ASCII 码是 0 ~ 255(8 位)。

- ◆ 停止位：用于表示单个包的最后一位，典型的值为 1，1.5 和 2 位。由于数是在传输线上定时的，并且每一个设备有其自己的时钟，很可能在通信中两台设备间出现了小小的不同步。因此停止位不仅仅是表示传输的结束，并且提供计算机校正时钟同步的机会。

- ◆ 奇偶校验位：在串口通信中一种简单的检错方式。对于偶和奇校验的情况，串口会设置校验位（数据位后面的一位），用一个值确保传输的数据有偶个或者奇

总线接口技术

个逻辑高位。例如，如果数据是 011，那么对于偶校验，校验位为 0，保证逻辑高的位数是偶数个。如果是奇校验，校验位为 1，这样就有 3 个逻辑高位。

④ 串口通信的传输格式：

串行通信中，线路空闲时，线路的 TTL 电平总是高，经反向 RS232 的电平总是低。一个数据的开始 RS232 线路为高电平，结束时 Rs232 为低电平。数据总是从低位向高位一位一位的传输。示波器读数时，左边是数据的高位。

例如，对于 16 进制数据 55aaH，当采用 8 位数据位、1 位停止位传输时，它在信号线上的波形如图 1(TTL 电平)和图 2(RS-232 电平)所示。

55H=01010101B，取反后 10101010B，加入一个起始位 1，一个停止位 0，55H 的数据格式为 1010101010B;aaH=10101010B，取反后 01010101B，加入一个起始位 1，一个停止位 0，55H 的数据格式为 1101010100B;

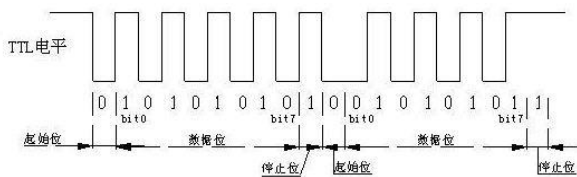


图1 TTL电平的串行数据帧格式(55aaH)

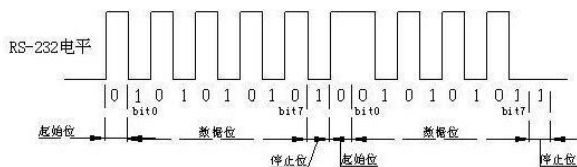


图2 RS-232电平的串行数据帧格式(55aaH)

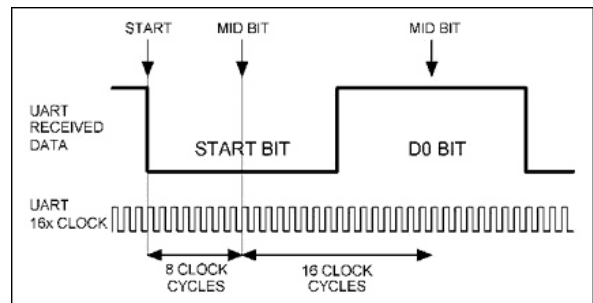
⑤ 串口通信的接收过程：

(异步通信：接收器和发送器有各自的时钟；同步通信：发送器和接收器由同一个时钟源控制。RS232 是异步通信)

- ◆ 开始通信时，信号线为空闲 (逻辑 1)，当检测到由 1 到 0 的跳变时，开始对“接收时钟”计数。
- ◆ 当计到 8 个时钟时，对输入信号进行检测，若仍为

低电平，则确认这是“起始位”，而不是干扰信号。

- ◆ 接收端检测到起始位后，隔 16 个接收时钟，对输入信号检测一次，把对应的值作为 D0 位数据。若为逻辑 1，作为数据位 1；若为逻辑 0，作为数据位 0。
- ◆ 再隔 16 个接收时钟，对输入信号检测一次，把对应的值作为 D1 位数据。....，直到全部数据位都输入。
- ◆ 检测校验位 P(如果有的话)。
- ◆ 接收到规定的数据位个数和校验位后，通信接口电路希望收到停止位 S(逻辑 1)，若此时未收到逻辑 1，说明出现了错误，在状态寄存器中置“帧错误”标志。若没有错误，对全部数据位进行奇偶校验，无校验错时，把数据位从移位寄存器中送数据输入寄存器。若校验错，在状态寄存器中置奇偶错标志。
- ◆ 本帧信息全部接收完，把线路上出现的高电平作为空闲位。
- ◆ 当信号再次变为低时，开始进入下一帧的检测。



单片机常用 11.0592M 的晶振，这个奇怪数字是有来历的：波特率为 9600BPS 每位位宽 $t_1=1/9600s$ ，晶振周期 $t_2=1/11.0592/1000000S$ ，单片机机器周期 $t_3=12*t_2$

$t_1/t_3=96$ 。即对于 9600BPS 的串口，单片机对其以 96 倍的速率进行采样。如果单片机晶振用的不正确，会对串口接受产生误码。

⑥ RS485 和 RS422

RS121 虽然使用很广，但由于推出时间比较早，所以在现代通信网络中已暴露明显的缺点，主要表现在：

- ◆ 接口的信号电平值较高，易损坏接口电路的芯片，又因为与 TTL 电平不兼容故需使用电平转换电路方能

总线接口技术

与 TTL 电路连接。

◆ 传输速率较低，在异步传输时，波特率为 20Kbps。现在由于采用新的 UART 芯片 16C550 等，波特率达到 115.2 Kbps。

◆ 接口使用一根信号线和一根信号返回线而构成共地的传输形式，这种共地传输容易产生共模干扰，所以抗噪声干扰性弱。

◆ 传输距离有限，最大传输距离标准值为 50 米，实际上也只能用在 15 米左右。因此在 RS232 的基础上提出了 RS485 和 RS422 总线。

RS485 总线，在要求通信距离为几十米到上千米时，广泛采用 RS-485 串行总线标准。RS485 采用平衡发送和差分接收，因此具有抑制共模干扰的能力。加上总线收发器具有高灵敏度，能检测低至 200mV 的电压，故传输信号能在千米以外得到恢复。RS485 采用半双工工作方式，任何时候只能有一点处于发送状态，因此，发送电路须由使能信号加以控制。RS485 用于多点互连时非常方便，可以省掉许多信号线。应用 RS485 可以联网构成分布式系统，其允许最多并联 32 台驱动器和 32 台接收器。

RS422 总线，RS485 和 RS422 电路原理基本相同，都是以差动方式发送和接受，不需要数字地线。差动工作是同速率条件下传输距离远的根本原因，这正是二者与 RS232 的根本区别，因为 RS232 是单端输入输出，双工工作时至少需要数字地线发送线和接受线三条线（异步传输），还可以加其它控制线完成同步等功能。RS422 通过两对双绞线可以全双工工作收发互不影响，而 RS485 只能半双工工作，收发不能同时进行，但它只需要一对双绞线。RS422 和 RS485 在 19kbps 下能传输 1200 米。用新型收发器线路上可连接台设备。

(5) USB 总线

① USB 简介

USB (Universal Serial Bus) 通用串行总线是由 Compaq、Digital Equipment、Intel、Microsoft、IBM、NEC 及 Northern Telecom 等 7 家公司联合开发的一种流行的外设接口标准。

USB 有着很广泛的应用：USB2.0 标准的 USB 硬盘、扫描仪和数码相机、数码摄像机等需大数据量传输

的新产品已投入使用；出现了 USB 转接设备，提供 USB 接口到其他接口的转换，这样，非 USB 接口的外设就能间接地使用 USB 接口，如 USB 到 PS / 2、USB 到 PCI 等；使用 USB 接口可是笔记本电脑的外设连接变得方便简单，机箱上提供多个 USB 接口，USB 光驱、USB 软驱等作为配件，极大地方便了用户使用。而且系统结构简单，可以使更高主频的微处理器用于笔记本电脑中。

② USB 的性能特点与基本特性

◆ USB 的性能特点

a. 传输速度快

USB 低速 1.5 Mbps，低速的 USB 支持低速设备，例如，调制解调器、键盘、鼠标、U 盘、硬盘、光驱、网卡、扫描仪、数码相机等；

USB 全速 12 Mbps，USB 全速的数据传输用于大范围的多媒体设备。

USB2.0 高速数据传输速度可以高达 480 Mbps。

b. 连接简单快捷

可以“热插拔”，真正的“即插即用”。设备连接到 USB 时，不必打开机箱，也不必关闭主机电源。USB 自检测，软件自动配置，完成后立刻就能使用。

c. 通用连接器

USB 用一种通用的连接器可以连接多种类型的外设，其外形为 4 针插头。



mini USB(A型)、mini USB (B型)、USB (B型)、USB (A型)、USB (A型)

A 型连接头用于固定的外围设备；B 型连接头用于经常插拔的设备。

d. 无须外接电源，数据线供电

由 USB 总线提供电源到外部设备，USB 能提供 + 5V / 500mA 的电源，供低功耗 USB 设备如 USB 键盘、USB 鼠标等作电源使用。高功耗的 USB 设备，如扫

总线接口技术

扫描仪等仍需自带电源；

USB 还采用 APM (Advanced Power Management) 技术，可以有效地节省电源功耗。如果 3ms 没有总线活动的话，USB 会自动进入挂起状态。进入挂起状态的设备消耗的电流不超过 500 μ A。

e. 扩充外设能力强

USB 采用星形层式结构和 Hub 技术，允许一个 USB 主控机可以连接多达 127 个外设，用户不用担心要连接的设备数目会受到限制。两个外设间的距离可达 5 米，扩充方便。

f. 不需要系统资源

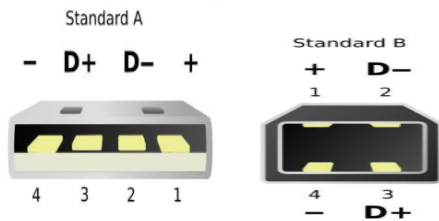
USB 设备不占用内存或 I/O 地址空间，而且也占用 IRQ 和 DMA 通道，所有事物都是由 USB 主机管理。

g. 错误检测和恢复

USB 事务处理包括错误保护机制，确保数据无错误发送。在发生错误时，事务处理可以重来。

◆ USB 的物理接口和电器特性

a. USB 接口定义：



b. USB 引脚定义如下：

连接序号	信号名称	典型连接线
1	VBUS (USB电源)	红
2	D+ (正差分信号)	白
3	D- (负差分信号)	绿
4	GND (USB地)	黑
外层	屏蔽层	—

c. USB 的电气特性

收发器：对地电源电压为 4.75 ~ 5.25V，设备吸入的

最大电流值为 500mA，D+、D- 上不加电压。

USB 设备：高速在 D+ 上加 3.0 ~ 3.6V 电压，低速反之。

D+、D- 线电气特性：

无驱动：高速 $V_{D+} > 2.7V, V_{D-} < 0.8V$ ，低速反之：

有驱动：高速 $V_{D+} > 2.0V, V_{D-} < 2.0V$ ，低速反之：

D+、D- 线信号特性：

高电平： $V_{D+} - V_{D-} > 200mV, V_{D+} > 2.0V$ ；

低电平： $V_{D-} - V_{D+} > 200mV, V_{D-} > 2.0V$ 。

D+、D- 线数据特性：NRZI (不归零反向码)

逻辑 0: D+、D- 线上电平高、低间跳边；

逻辑 1: D+、D- 线上电平不变。

d. 电源

USB 系统可以通过 USB 线缆为其外设提供不高于 +5V、500mA 的总线电源。

有两类器件定义如下：

- 完全依靠 USB 线缆来提供电源的器件被称为总线驱动器件。

- 自带电源的器件则被称为自驱动外设。

当一个外设初次连接时，器件的配置和分类并不使用外设自带的电源，而是通过 USB 线缆提供的电源来使外设处于 Powered 状态。

e. 电源管理系统

设备的挂起状态：当外设长时间 (3.0ms 以上) 处于非使用状态时，USB 系统自动将该设备挂起，当一个 USB 外设处于挂起状态时，USB 总线通过 USB 线缆为该设备仅仅提供 500 μ A 以下的电流，并把该外设所占用的 USB 带宽分配给其它的 USB 外设。

唤醒：当外设处于挂起状态时，必须先通过主机使该设备“唤醒”，然后才可以执行 USB 操作。

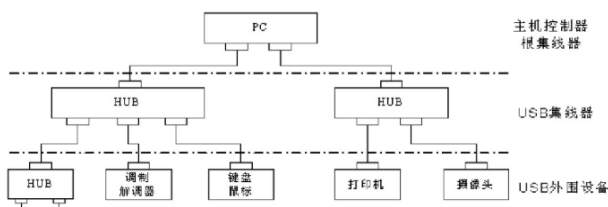
总线接口技术

f. USB 总线系统中的设备

USB 总线系统中的设备可以分为三个类型：

- USB 主机：只能有一个，管理 USB 系统，每秒产生一帧数据；发送配置请求对 USB 设备进行配置操作，对总线上的错误进行管理和恢复。
- USB 集线器 (HUB)：类似于网络集线器，完成 USB 设备的添加 (扩展)、删除和电源管理等。
- USB 总线设备 (USB 功能外设)

组成金字塔形结构，如下图所示，最多可连接 7 层，127 台设备。



③ USB 的传输方式

USB 有 4 种基本的传输方式：

◆ 批处理方式

批处理可以是单向，也可以是双向。它用于传输大批数据，这种数据传输的时间性不强，但要确保数据的正确性。在数据包的传输过程中，出现错误，则需重新传输。其典型的应用是扫描仪、打印机和数码相机。

◆ 中断传输

中断传输是单向的，且仅输入到主机，它用于不固定的、少量的数据传输，但这些数据需要及时处理。当设备需要主机为其服务时，想主机发送此类信息以通知主机，像键盘、鼠标之类的输入设备采用这种方式。USB 的中断传输是 Polling (查询) 类型。主机要频繁地请求端点输入。USB 设备在全速情况下，其端点 Polling 周期为 1 ~ 255 ms。因此，最快的 Polling 频率是 1 KHz，在信息的传输过程中，如果出现错误，则需在下一个 Polling 中重新传输。

◆ 等时传输

等时传输也称同步传输，可以单向也可以双向，用于

传输连续性、实时的数据。这种方式的特点是要求传输速率固定，时间性强，忽略传输错误，即传输中出错也不重传。因为这样会影响传输速率。传输的最大数据包是 1.24 B/ms。如视频设备、数字声音设备采用这种方式。

◆ 控制传输

控制传输是双向的，它的传输有 2 ~ 3 个阶段：Setup 阶段 (可以没有)，Data 阶段和 Status 阶段。在 Setup 阶段，主机传送命令给设备；在 Data 阶段，传输的是 Setup 阶段所设定的数据；Status 阶段，设备返回握手信号给主机。

(6) IEEE 488 总线

随着微机技术的普及发展，人们建立的现代实验与测试系统大多以微机为核心，形成有智能色彩的自动测试系统。这里，数据通道成了各仪器设备之间的桥梁与关键。通常人们使用 RS-232 或 IEEE-488 总线来构成数据通路。而 RS-232 属于串行接口，数据只能以 Bit 的方式进行传输，波特率低，在许多要求高速传输的场合达不到预想的效果。因此以 Byte 方式传输的 IEEE-488 并行接口总线就为许多系统所采用。

IEEE-488 接口总线是个工业标准总线，它是由国际电子电气工程师协会于 1974 年 9 月制定的一种标准接口系统。具有以下六个要素：

◆ 在有限的距离内使用。

◆ 具有广泛的通用性，可以将不同工厂独立制造且不同用途的各种产品互相连接，组成一个工作系统。

◆ 允许各被连接的装置之间直接进行通讯联系，而不一定非得经过一个控者单元在能进行。

◆ 通信应是双向异步的，装置的数据速率应允许在广泛的范围内变化。

◆ 对那些被连接的装置，应提出尽可能少的限制条件。

◆ 成本低廉，使用方便。

① 总线的主要特征：

◆ 总线型连接方式

总线接口技术

测控系统内计算机和所有仪器通过一组标准总线互相连接，系统的连接方式比较方便、自由。配有标准接口的仪器可以比较方便地接入系统，仪器数量的增减不影响其他仪器的连接。组建系统时，不论采用什么样的连接方式，只需用标准总线将仪器连接起来，解散系统时也只需讲连接仪器的总线拔出，各台仪器又可以单独使用。

◆ 总线构成

IEEE-488 总线包括 16 条信号线，其中 8 条数据线、5 条接口管理线和 3 条挂钩线。总线采用 24 脚插头座。

◆ 器件容量

包括计算机在内，系统内可连接的设备数目最多 15 个，这主要受目前 TTL 接口驱动器最大驱动电流的影响。当组建测试系统所需要的器件数目多余 15 个时，只能在计算机内再添加一个 IEEE 488 接口。通过这个接口可以再连接 14 个器件。

◆ 地址容量

系统中每一个仪器都设置有一个地址，IEEE 488 协议规定地址为 5 位，可得到 $2^5=32$ 个地址。

◆ 数据传输方式

IEEE - 488 总线是一种异步双向型总线，按照位并行和字节串行方式传输数据，信号传输速率为 1Mbps（高速 HSIB 为 10Mbps）

◆ 传输距离

整个系统的连接电缆最大长度为 20m。

◆ 接口功能

仪器和接口系统之间的每一种交互作用称为一种接口功能，IEEE-488 系统中共设立了 10 中接口功能，如听者功能、讲者功能、控者功能等。

◆ 工作方式

“讲者”是通过总线发送仪器消息的仪器装置（如测量仪器、数据采集器、计算机等），一个系统中在同一时刻，只能有一个讲者在起作用。

“听者”是通过总线接受由讲者发出消息的装置（如打印机、信号源等），在一个系统中，可以设置多个听者，

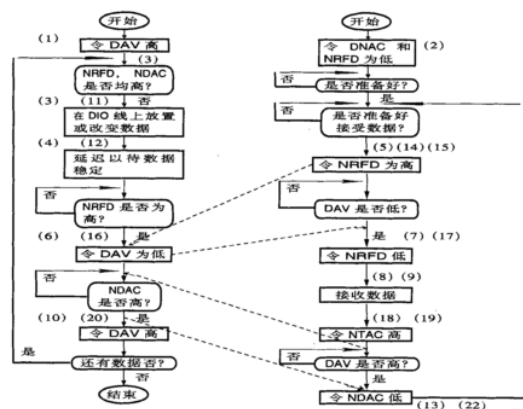
并且允许多个听者同时工作。

“控者”是数据传输过程中的组织者和控制者，例如对其他设备进行寻址或允许“讲者”使用总线等。控者通常由计算机担任。

② 接口系统的母线结构和接口功能

IEEE-488 标准将与接口问题有关的部分和装置基本工作特性部分严格区分开，以求得到最大的灵活性和兼容性。IEEE-488 系统具体地规定了接口的三大要素——机械的、电气的和功能的有关方面 通过对接口三大要素的规定 保证了装置连接的兼容性。由于将接口功能的装置功能作了严格的区分，仪器制造部门可以自由地设计实现各种特性的仪器装置，而对装置的接口部分 IEEE-488 标准只规定了逻辑关系，所以使设计接口电路也具有很大的灵活性，使兼容性与灵活性之间有了最大限度的统一，对装置提出的限制条件也减到最少。

IEEE-488 母线是一条 24 芯的无源电缆线，其中十六条被用作信号线。其余则被用作逻辑地或外屏蔽。电缆的内部结构 应保证各信号线之间的串漏、外界信号对信号线的影响以及信号线的信号向外界的辐射等效应都抑制到最小。为此，将若干信号线和逻辑地线依次或采用等效的方法使之隔离。同时，电缆外侧必须有一个总屏蔽层，这个总屏蔽层的编辑覆盖系数至少为 85%。电缆中各信号线每米最大电阻为 0.14Ω 。公共逻辑地线每米最大电阻为 0.085Ω 。任一条信号线与其它全部导线（包括信号线、逻辑地和外屏蔽）之间在一千赫兹频率测得的最大电容（当其它导线全部接地时）应不大于 150 pf/m 。



总线接口技术

由于 GPIB 总线是基于 IEEE-488 的的标准建立的, IEEE-488 总线的结构及其三线挂钩的传输方式在 GPIB 总线接口部分已经介绍过, 在此不再赘述。

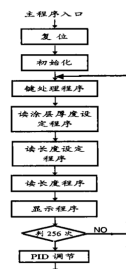
③ 基于 IEEE-488 总线的系统结构

本例子是在和核环境下的超声波探测与试验系统, 为了建立符合 IEEE-488 标准的测试系统, 采用型号为 NEC-7210 的大规模集成电路研制了一块 PC 机用 IEEE-488 标准接口板, 在此系统中将一台 PC 机作为系统控者, 将一台超声分析仪、一台数字存储示波器及一台绘图仪最为听者与讲者 (或只听), 系统的硬件结构如下图, 这里超声分析仪、数字存储示波器及绘图仪本身都带有 IEEE-488 标准接口, 通过 IEEE-488 标准接口线将它们于 PC 机相连。超声分析仪通过换能器发射与接收高频超声波, 对被测件进行探测, 并作初步的检波、阻抗变换等分析与处理。数字存储示波器对超声信号进行模数转换、图像压缩等。绘图仪则将结果绘出。PC 机既作为系统控者、又作为超声信号处理与保存者。



IEEE-488 十大接口功能的给出, 使得可以用程序来控制 IEEE-488 接口中消息的传递。但是, 在编制有关软件时, 从编程语言的选择到实时性安全性的统一等等问题, 都是需要考虑的。同时, 对于 IEEE-488 接口来讲, 它本身的输入、输出细节已被库函数所隐藏。实际工作中我们选用 C 语言来编写软件。

主程序框图如下:



主程序框图

对于不同的操作以及不同的使用环境都有良好的实用能力, 这就要求系统能设置较多的参数以供选用。

在软件运行中, 系统不断地检查 IEEE-488 卡的返回值 STA-TUS (整数型指针), 观察其是否为“-1”, 以确定 IEEE-488 卡的工作情况 (“-1”为异常)。而且 IEEE-488 卡的初始化和控制与机器周期比较要慢的多。此时, 系统就进行大量的数据处理与判断工作, 来节省 CPU 时间开销, 加快运行速度。在软件调试过程中发现: 当计算机要求 IEEE-488 接口准备数据时, 此时各 DIO 线上的数据字节尚未建立; 而计算机的执行速度又很快, 若此时采集数据, 容易造成数据漏采或是采集到错误数据的情况。这就需要在程序中加入一点延时。本测试系统已得到实际应用。

(7) CAN 总线

① CAN 总线的主要特征

CAN 是 Controller Area Network 的缩写 (以下称为 CAN), 是 ISO 国际标准化的串行通信协议。在当前的汽车产业中, 出于对安全性、舒适性、方便性、低公害、低成本的要求, 各种各样的电子控制系统被开发了出来。由于这些系统之间通信所用的数据类型及对可靠性的要求不尽相同, 由多条总线构成的情况很多, 线束的数量也随之增加。为适应“减少线束的数量”、“通过多个 LAN, 进行大量数据的高速通信”的需要, 1986 年德国电气商博世公司开发出面向汽车的 CAN 通信协议。此后, CAN 通过 ISO11898 及 ISO11519 进行了标准化, 现在在欧洲已是汽车网络的标准协议。

现在, CAN 的高性能和可靠性已被认同, 并被广泛地应用于工业自动化、船舶、医疗设备、工业设备等方面。现场总线是当今自动化领域技术发展的热点之一, 被誉为自动化领域的计算机局域网。它的出现为分布式控制系统实现各节点之间实时、可靠的数据通信提供了强有力的技术支持。

CAN 属于现场总线的范畴, 它是一种有效支持分布式控制或实时控制的串行通信网络。较之目前许多 RS-485 基于 R 线构建的分布式控制系统而言, 基于 CAN 总线的分布式控制系统在以下方面具有明显的优越性:

首先, CAN 控制器工作于多主方式, 网络中的各节点都可根据总线访问优先权 (取决于报文标识符) 采用

总线接口技术

无损结构的逐位仲裁的方式竞争向总线发送数据，且 CAN 协议废除了站地址编码，而代之以对通信数据进行编码，这可使不同的节点同时接收到相同的数据，这些特点使得 CAN 总线构成的网络各节点之间的数据通信实时性强，并且容易构成冗余结构，提高系统的可靠性和系统的灵活性。而利用 RS-485 只能构成主从式结构系统，通信方式也只能以主站轮询的方式进行，系统的实时性、可靠性较差；

其次，CAN 总线通过 CAN 收发器接口芯片 82C250 的两个输出端 CANH 和 CANL 与物理总线相连，而 CANH 端的状态只能是高电平或悬浮状态，CANL 端只能是低电平或悬浮状态。这就保证不会出现现象在 RS-485 网络中，当系统有错误，出现多节点同时向总线发送数据时，导致总线呈现短路，从而损坏某些节点的现象。而且 CAN 节点在错误严重的情况下具有自动关闭输出功能，以使总线上其他节点的操作不受影响，从而保证不会出现现象在网络中，因个别节点出现问题，使得总线处于“死锁”状态。而且，CAN 具有的完善的通信协议可由 CAN 控制器芯片及其接口芯片来实现，从而大大降低系统开发难度，缩短了开发周期，这些是只仅仅有电气协议的 RS-485 所无法比拟的。

另外，与其它现场总线比较而言，CAN 总线是具有通信速率高、容易实现、且性价比高等诸多特点的一种已形成国际标准的现场总线。这些也是目前 CAN 总线应用于众多领域，具有强劲的市场竞争力的重要原因。

CAN 即控制器局域网络，属于工业现场总线的范畴。与一般的通信总线相比，CAN 总线的数据通信具有突出的可靠性、实时性和灵活性。由于其良好的性能及独特的设计，CAN 总线越来越受到人们的重视。它在汽车领域上的应用是最广泛的，世界上一些著名的汽车制造厂商，如 BENZ(奔驰)、BMW(宝马)、PORSCHE(保时捷)、ROLLS-ROYCE(劳斯莱斯)和 JAGUAR(美洲豹)等都采用了 CAN 总线来实现汽车内部控制系统与各检测和执行机构间的数据通信。同时，由于 CAN 总线本身的特点，其应用范围目前已不再局限于汽车行业，而向自动控制、航空航天、航海、过程工业、机械工业、纺织机械、农用机械、机器人、数控机床、医疗器械及传感器等领域发展。CAN 已经形成国际标准，并已被公认为几种最有前途的现场总线之一。其典型的应用协议有：SAE J1939/ISO11783、CANOpen、CANaerospace、

DeviceNet、NMEA 2000 等。

② CAN 总线的产生与发展

控制器局部网 (CAN - CONTROLLER AREA NETWORK) 是 BOSCH 公司为现代汽车应用领先推出的一种多主机局部网，由于其高性能、高可靠性、实时性等优点现已广泛应用于工业自动化、多种控制设备、交通工具、医疗仪器以及建筑、环境控制等众多部门。控制器局部网将在我国迅速普及推广。

随着计算机硬件、软件技术及集成电路技术的迅速发展，工业控制系统已成为计算机技术应用领域中最具活力的一个分支，并取得了巨大进步。由于对系统可靠性和灵活性的高要求，工业控制系统的发展主要表现为：控制面向多元化，系统面向分散化，即负载分散、功能分散、危险分散和地域分散。

分散式工业控制系统就是为适应这种需要而发展起来的。这类系统是以微型机为核心，将 5C 技术 -- COMPUTER(计算机技术)、CONTROL(自动控制技术)、COMMUNICATION(通信技术)、CRT(显示技术)和 CHANGE(转换技术)紧密结合的产物。它在适应范围、可扩展性、可维护性以及抗故障能力等方面，较之分散型仪表控制系统和集中型计算机控制系统都具有明显的优越性。

典型的分散式控制系统由现场设备、接口与计算设备以及通信设备组成。现场总线 (FIELD BUS) 能同时满足过程控制和制造业自动化的需要，因而现场总线已成为工业数据总线领域中最活跃的一个领域。现场总线的研究与应用已成为工业数据总线领域的热点。尽管目前对现场总线的研究尚未能提出一个完善的标准，但现场总线的高性能价格比将吸引众多工业控制系统采用。同时，正由于现场总线的标准尚未统一，也使得现场总线的应用得以不拘一格地发挥，并将为现场总线的完善提供更加丰富的依据。控制器局部网 CAN (CONTROLLER AREA NETWORK) 正是在这种背景下应运而生的。

由于 CAN 为愈来愈多不同领域采用和推广，导致要求各种应用领域通信报文的标准化。为此，1991 年 9 月 PHILIPS SEMICONDUCTORS 制订并发布了 CAN 技术规范 (VERSION 2.0)。该技术规范包括 A 和 B 两部分。2.0A 给出了曾在 CAN 技术规范版本 1.2

总线接口技术

中定义的 CAN 报文格式，能提供 11 位地址；而 2.0B 给出了标准的和扩展的两种报文格式，提供 29 位地址。此后，1993 年 11 月 ISO 正式颁布了道路交通运载工具 -- 数字信息交换 -- 高速通信控制器局部网 (CAN) 国际标准 (ISO11898)，为控制器局部网标准化、规范化推广铺平了道路。

③ CAN 总线特点

CAN 总线是德国 BOSCH 公司从 80 年代初为解决现代汽车中众多的控制与测试仪器之间的数据交换而开发的一种串行数据通信协议，它是一种多主总线，通信介质可以是双绞线、同轴电缆或光导纤维。通信速率可达 1MBPS。

◆ CAN 总线通信接口中集成了 CAN 协议的物理层和数据链路层功能，可完成对通信数据的成帧处理，包括位填充、数据块编码、循环冗余检验、优先级判别等工作。

◆ CAN 协议的一个最大特点是废除了传统的站地址编码，而代之以对通信数据块进行编码。采用这种方法的优点可使网络内的节点个数在理论上不受限制，数据块的标识码可由 11 位或 29 位二进制数组成，因此可以定义 2¹¹ 或 2²⁹ 个不同的数据块，这种按数据块编码的方式，还可使不同的节点同时接收到相同的数据，这一点在分布式控制系统中非常有用。数据段长度最多为 8 个字节，可满足通常工业领域中控制命令、工作状态及测试数据的一般要求。同时，8 个字节不会占用总线时间过长，从而保证了通信的实时性。CAN 协议采用 CRC 检验并提供相应的错误处理功能，保证了数据通信的可靠性。CAN 卓越的特性、极高的可靠性和独特的设计，特别适合工业过程监控设备的互连，因此，越来越受到工业界的重视，并已公认为最有前途的现场总线之一。

◆ CAN 总线采用了多主竞争式总线结构，具有多主站运行和分散仲裁的串行总线以及广播通信的特点。CAN 总线上任意节点可在任意时刻主动地向网络上其它节点发送信息而不分主次，因此可在各节点之间实现自由通信。CAN 总线协议已被国际标准化组织认证，技术比较成熟，控制的芯片已经商品化，性价比高，特别适用于分布式测控系统之间的数通讯。CAN 总线插卡可以任意插在 PC AT XT 兼容机上，方便地构成分布式监控系统。

◆ 结构简单，只有 2 根线与外部相连，并且捏布集成了错误探测和管理模块。

④ CAN 总线技术介绍

◆ 位仲裁

要对数据进行实时处理，就必须将数据快速传送，这就要求数据的物理传输通路有较高的速度。在几个站同时需要发送数据时，要求快速地进行总线分配。实时处理通过网络交换的紧急数据有较大的不同。一个快速变化的物理量，如汽车引擎负载，将比类似汽车引擎温度这样相对变化较慢的物理量更频繁地传送数据并要求更短的延时。

CAN 总线以报文为单位进行数据传送，报文的优先级结合在 11 位标识符中，具有最低二进制数的标识符有最高的优先级。这种优先级一旦在系统设计时被确立后就不能再被更改。总线读取中的冲突可通过位仲裁解决。如图 2 所示，当几个站同时发送报文时，站 1 的报文标识符为 011111；站 2 的报文标识符为 0100110；站 3 的报文标识符为 0100111。所有标识符都有相同的两位 01，直到第 3 位进行比较时，站 1 的报文被丢掉，因为它的第 3 位为高，而其它两个站的报文第 3 位为低。站 2 和站 3 报文的 4、5、6 位相同，直到第 7 位时，站 3 的报文才被丢失。注意，总线中的信号持续跟踪最后获得总线读取权的站的报文。在此例中，站 2 的报文被跟踪。这种非破坏性位仲裁方法的优点在于，在网络最终确定哪一个站的报文被传送以前，报文的起始部分已经在网络上传送了。所有未获得总线读取权的站都成为具有最高优先权报文的接收站，并且不会在总线再次空闲前发送报文。

CAN 具有较高的效率是因为总线仅仅被那些请求总线悬而未决的站利用，这些请求是根据报文在整个系统中的重要性按顺序处理的。这种方法在网络负载较重时有很多优点，因为总线读取的优先级已被按顺序放在每个报文中了，这可以保证在实时系统中较低的个体潜伏时间。

对于主站的可靠性，由于 CAN 协议执行非集中化总线控制，所有主要通信，包括总线读取（许可）控制，在系统中分几次完成。这是实现有较高可靠性的通信系统的唯一方法。

总线接口技术

◆ CAN 与其它通信方案的比较

在实践中,有两种重要的总线分配方法:按时间表分配和按需要分配。在第一种方法中,不管每个节点是否申请总线,都对每个节点按最大期间分配。由此,总线可被分配给每个站并且是唯一的站,而不论其是立即进行总线存取或在一特定时间进行总线存取。这将保证在总线存取时有明确的总线分配。在第二种方法中,总线按传送数据的基本要求分配给一个站,总线系统按站希望的传送分配(如:Ethernet CSMA/CD)。因此,当多个站同时请求总线存取时,总线将终止所有站的请求,这时将不会有任一个站获得总线分配。为了分配总线,多于一个总线存取是必要的。

CAN 实现总线分配的方法,可保证当不同的站申请总线存取时,明确地进行总线分配。这种位仲裁的方法可以解决当两个站同时发送数据时产生的碰撞问题。不同于 Ethernet 网络的消息仲裁,CAN 的非破坏性解决总线存取冲突的方法,确保在不传送有用消息时总线不被占用。甚至当总线在重负载情况下,以消息内容为优先的总线存取也被证明是一种有效的系统。虽然总线的传输能力不足,所有未解决的传输请求都按重要性顺序来处理。在 CSMA/CD 这样的网络中,如 Ethernet,系统往往由于过载而崩溃,而这种情况在 CAN 中不会发生。

◆ CAN 的报文格式

在总线中传送的报文,每帧由 7 部分组成。CAN 协议支持两种报文格式,其唯一的不同是标识符(ID)长度不同,标准格式为 11 位,扩展格式为 29 位。

在标准格式中,报文的起始位称为帧起始(SOF),然后是由 11 位标识符和远程发送请求位(RTR)组成的仲裁场。RTR 位标明是数据帧还是请求帧,在请求帧中没有数据字节。

控制场包括标识符扩展位(IDE),指出是标准格式还是扩展格式。它还包括一个保留位(ro),为将来扩展使用。它的最后四个字节用来指明数据场中数据的长度(DLC)。数据场范围为 0~8 个字节,其后有一个检测数据错误的循环冗余检查(CRC)。

应答场(ACK)包括应答位和应答分隔符。发送站发送的这两位均为隐性电平(逻辑 1),这时正确接收报文的接收站发送主控电平(逻辑 0)覆盖它。用这种方法,

发送站可以保证网络中至少有一个站能正确接收到报文。

报文的尾部由帧结束标出。在相邻的两条报文间有一很短的间隔位,如果这时没有站进行总线存取,总线将处于空闲状态。

◆ 数据错误检测

不同于其它总线,CAN 协议不能使用应答信息。事实上,它可以将发生的任何错误用信号发出。CAN 协议可使用五种检查错误的方法,其中前三种为基于报文内容检查。

a. 循环冗余检查(CRC)

在一帧报文中加入冗余检查位可保证报文正确。接收站通过 CRC 可判断报文是否有错。

b. 帧检查

这种方法通过位场检查帧的格式和大小来确定报文的正确性,用于检查格式上的错误。

c. 应答错误

如前所述,被接收到的帧由接收站通过明确的应答来确认。如果发送站未收到应答,那么表明接收站发现帧中有错误,也就是说,ACK 场已损坏或网络中的报文无站接收。CAN 协议也可通过位检查的方法探测错误。

d. 总线检测

有时,CAN 中的一个节点可监测自己发出的信号。因此,发送报文的站可以观测总线电平并探测发送位和接收位的差异。

e. 位填充

一帧报文中的每一位都由不归零码表示,可保证位编码的最大效率。然而,如果在一帧报文中有多相同电平的位,就有可能失去同步。为保证同步,同步沿位填充产生。在五个生。在五个连续相等位后,发送站自动插入一个与之互补的补码位;接收时,这个填充位被自动丢掉。例如,五个连续的低电平位后,CAN 自动插入一个高电平位。CAN 通过这种编码规则检查错误,如果在一帧报文中有多 6 个相同位,CAN 就知道发生了错误。

总线接口技术

如果至少有一个站通过以上方法探测到一个或多个错误，它将发送出错标志终止当前的发送。这可以阻止其它站接收错误的报文，并保证网络上报文的一致性。当大量发送数据被终止后，发送站会自动地重新发送数据。作为规则，在探测到错误后 23 个位周期内重新开始发送。在特殊场合，系统的恢复时间为 31 个位周期。

但这种方法存在一个问题，即一个发生错误的站将导致所有数据被终止，其中也包括正确的数据。因此，如果不采取自监测措施，总线系统应采用模块化设计。为此，CAN 协议提供一种将偶然错误从永久错误和局部站失败中区别出来的办法。这种方法可以通过对出错站统计评估来确定一个站本身的错误并进入一种不会对其它站产生不良影响的运行方法来实现，即站可以通过关闭自己来阻止正常数据因被错误地当成不正确的数据而被终止。

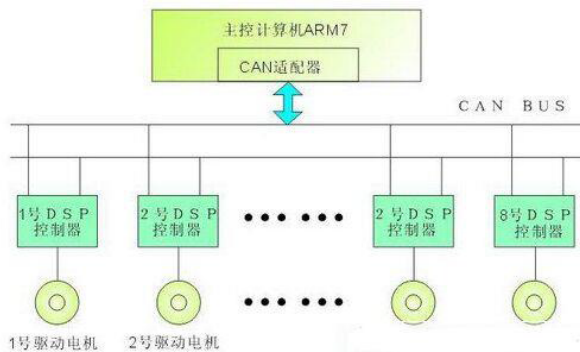
e. CAN 可靠性

为防止汽车在使用寿命期内由于数据交换错误而对司机造成危险，汽车的安全系统要求数据传输具有较高的安全性。如果数据传输的可靠性足够高，或者残留下来的数据错误足够低的话，这一目标不难实现。从总线系统数据的角度看，可靠性可以理解为，对传输过程产生的数据错误的识别能力。

残余数据错误的概率可以通过对数据传输可靠性的统计测量获得。它描述了传送数据被破坏和这种破坏不能被探测出来的概率。残余数据错误概率必须非常小，使其在系统整个寿命周期内，按平均统计时几乎检测不到。计算残余错误概率要求能够对数据错误进行分类，并且数据传输路径可由一模型描述。如果要确定 CAN 的残余错误概率，我们可将残留错误的概率作为具有 80 ~ 90 位的报文传送时位错误概率的函数，并假定这个系统中有 5 ~ 10 个站，并且错误率为 1/1000，那么最大位错误概率为 10—13 数量级。例如，CAN 网络的数据传输率最大为 1Mbps，如果数据传输能力仅使用 50%，那么对于一个工作寿命 4000 小时、平均报文长度为 80 位的系统，所传送的数据总量为 9×10^{10} 。在系统运行寿命期内，不可检测的传输错误的统计平均小于 10—2 量级。换句话说，一个系统按每年 365 天，每天工作 8 小时，每秒错误率为 0.7 计算，那么按统计平均，每 1000 年才会发生一个不可检测的错误。

◆ CAN 总线应用实例

制造车间底层设备自动化，近几年仍是我国开展新技术研究和新技术应用工程及产品开发的主要领域，其市场需求不断增大且越发活跃，竞争也日益激烈。伴随着工业机器人的产业化，目前机器人系统的应用大多要求采用机器人生产方式，这就要求多台机器人能通过网络进行互联。随之而来的是，在实际生产过程中，这种连网的多机器人系统的调度、维护工作也变得尤为重要。制造车间底层电气装置联网是近几年内技术发展的重点。其电器装置包括有：运动控制器、基于微处理器的传感器、专用设备控制器等底层设备；在这些装置所构成的网络上另有车间级管理机、监控机或生产单元控制器等非底层装置。结合实际情况和要求，将机器人控制器视为运动控制器。



把 CAN 总线技术充分应用于现有的控制器当中，将可开发出高性能的多机器人生产线系统。利用现有的控制技术，结合 CAN 技术和通信技术，通过对现有的机器人控制器进行硬件改进和软件开发，并相应地开发出上位机监控软件，从而实现多台机器人的网络互联。最终实现基于 CAN 网络的机器人生产线集成系统。这样做的好处很多，例如实现单根电缆串接全部设备，节省安装维护开销；提高实时性，信息可共享；提高多控制器系统的检测、诊断和控制性能；通过离线的任务调度、作业的下载以及错误监控等技术，把一部分人从机器人工作的现场彻底脱离出来。

CAN 总线的数据通信具有突出的可靠性、实时性和灵活性。由于其良好的性能及独特的设计，CAN 总线越来越受到人们的重视，它在汽车领域上的应用是最广泛的。世界上一些著名的汽车制造厂商大都采用了 CAN 总线来实现汽车内部控制系统与各检测和执行机构间的数据通信。同时，由于 CAN 总线本身的特点，

总线接口技术

其应用范围目前已不再局限于汽车行业，而向自动控制、航空航天、航海、过程工业、机械工业、纺织机械、农用机械、机器人、数控机床、医疗器械及传感器等领域发展。CAN 已经形成国际标准，并已被公认为几种最有前途的现场总线之一。

(8) LIN 总线技术

LIN 本地互联网络是一种将开关、显示器、传感器及执行器等简单控制设备连接起来的串行通信网络，主要用于实现汽车中的分布式电子系统控制。因其主要目标是为汽车网络（如 CAN 总线）提供辅助功能，因此通常作为子网络，用于一些不需要诸如 CAN 总线的带宽和多功能的场合，比如智能传感器和制动装置之间的通信。使用 LIN 总线可大大节省成本，该低成本的串行通信模式和相应的开发环境已经由 LIN 协会制定成标准，为汽车制造商以及供应商在研发、应用电子产品方面降低成本。但是，LIN 的应用并不局限于汽车领域，在诸如工业控制领域也理所应当占有广阔的应用地位和前景。

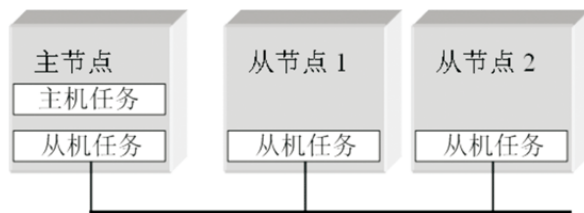
LIN 总线的技术特点包括：

- ◆ 单主机，多从机结构（没有总线仲裁）；
- ◆ 基于普通 UART/ SCI 接口的低成本硬件、低成本软件或作为纯状态机；
- ◆ 带时间同步的多点广播接收，从节点无需石英或陶瓷振荡器；
- ◆ 确定性的信号传输；
- ◆ 低成本的单线实现；
- ◆ 速率可达 20Kbit/s；
- ◆ 总线长度 $\leq 40\text{m}$ ；
- ◆ 保证信号传输的延迟时间；
- ◆ 可选的数据场长度 0~8bytes；
- ◆ 灵活性的配置；
- ◆ 数据校验和的安全性和错误检测；
- ◆ 网络中故障节点的检测；

- ◆ 使用最小成本的半导体元件（小尺寸，单芯片系统）；
- ◆ 不需改变 LIN 从节点的硬件和软件即可在网络上增加节点；
- ◆ 通常一个 LIN 网络节点数小于 16 个。

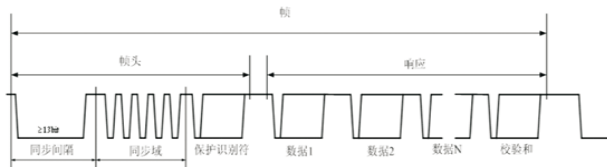
① LIN2.0 总线技术

LIN2.0 版本反映了 LIN 协会的定义趋势，通过比较 LIN1.3 和 LIN2.0 规范，可看到最重大的两个变化是对于配置和诊断的标准化支持，以及指定节点能力文件，这都是为了要简化现有节点的使用。LIN 工作原理是基于单主/多从概念。在一个 LIN 簇中，由一个主节点以及数个从节点构成。主节点由主机任务和从机任务组成。而所有其他的节点只包含从机任务。下图所示为典型的 LIN 簇，即一个主节点，两个从节点。主机任务决定何时以及传送哪一帧，而从机任务则传送帧数据。



单主多从原理图

LIN2.0 报文帧由帧头和帧响应组成，其中帧头由主机任务传送，帧响应由从机任务传送。每个报文帧都包含 2、4、8 字节的数据。报文帧的结构由一个同步间隔域 (Break) 以及随后的 4 至 11 个字节域 (byte field) 构成，如下图所示。



LIN 报文帧结构及字节域结构

总线接口技术

比较项目	LIN	CAN
最大通信速率	20kbit/s	1Mbit/s
媒体访问控制	无仲裁	由仲裁
最大通信距离	40m	10km
物理层	单总线	双绞线差动总线
典型网络大小	2-10节点	4-20节点
错误检测	8位校验	15位CRC

LIN 总线和 CAN 总线性能比较

◆ 同步间隔域

同步间隔域可看作是一中断，用来表示一个新报文帧的开始，由主节点中的主机任务在 LIN 总线上产生至少 13 位显性状态，其中包含起始位和同步界定符。

◆ 同步域：其格式为 0x55，表现为 8 个位定时中有 5 个下降沿。

◆ 保护识别符：用于描述报文数据的含义，一个保护识别符域由两个子域组成：其中第 0 位到第 5 位是识别符位，第 6 和第 7 位是奇偶校验位。识别符共 6 位，取值范围为 0 至 63。识别符可分为 4 类：

0 到 59(0x3b) 用于信号传输的报文帧；

60(0x3c) 和 61(0x3d) 用于传送诊断数据；

62(0x3e) 为留给用户的扩展帧；

63(0x3f) 为保留用于以后协议增订使用。

◆ 数据域：一个报文帧可以“携带”0 到 8 字节的数据。

◆ 校验和：校验和等于数据域所有字节的和的取反。对于只计算数据域所有字节的校验和，称为传统校验和。如果校验和要将保护识别符和数据域所有字节一起计算，则称为增强性校验和。

由表 1 可见，LIN 总线的整体性能要比 CAN 差很多，但 LIN 总线较 CAN 总线的最大优势在于实现成本比较低，由于 LIN 面向的是并不需要 CAN 的性能、带宽及复杂性的低端系统，因此在这些场合中 LIN 有较为广泛的应用。

② LIN 从节点应用设计

本书介绍 NXP 推出的本地互连网络 I/O 附属扩展芯片 UJA1023，可协助构建高性价比、高可靠度的车内 LIN 网络。虽然，该芯片起初的设计初衷是应用在高档轿车行业，但由于该芯片的高集成性和可靠性等特点，完全能够用于诸如电力工业领域等。UJA1023 是一个独立的可应用于本地互连网络 (LIN)/I/O 的从节点，它可以代替传统的电子控制单元中所使用的基本部件，这些电子控制单元用于对 I/O 进行控制。

UJA1023 芯片内包含一个 LIN 2.0 控制器，一个集成的 LIN 收发器，并同时兼容 LIN1.3 规范。同时还包括 8 个可通过 LIN 总线进行配置的 I/O 端口。UJA1023 具有自动比特率同步电路，可以在 1kbit/s 至 20kbit/s 范围内与主节点保持同步，因此芯片内集成了一个晶振。

LIN 协议由芯片自动进行处理，从节点的地址 NAD 和帧识别符 (frame Identifier) 的设置由主请求命令 (master request) 和从响应命令 (slave response) 以菊花链方式或即插方式完成。8 个双向 I/O 引脚可由 LIN 总线命令进行配置，并且可以具有以下功能：

a. 输入功能

- 标准输入引脚；
- 本地唤醒；
- 边沿捕获；
- 模拟信号输入引脚；
- 开关矩阵 (与输出引脚配合使用)。

b. 输出功能

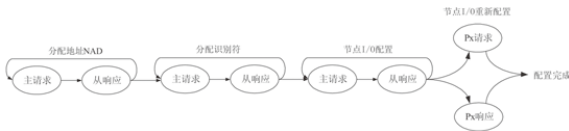
- 标准输出引脚，可配置为高边驱动、低边驱动、推拉驱动模式；
- 周期 (循环) 侦听模式用于本地唤醒；
- PWM 模式，例如，驱动轿车后灯；
- 开关矩阵 (与输入引脚配合使用)。

c. 配置流程：

UJA1023 可通过 LIN 命令帧，即主请求命令和从响

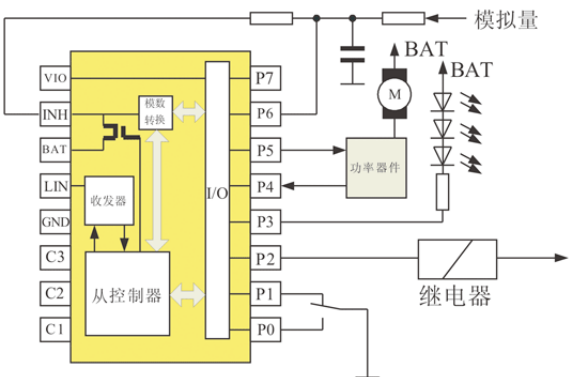
总线接口技术

应命令以报文链的形式完成配置。主请求命令发送配置数据，而主请求命令的目标从节点在收到命令后将发送相应的数据给主节点作为响应。下图所示为 UJA1023 的配置流程，其中主请求命令和从响应命令之间井然有序的配合就是一种“握手”的概念。



UJA1023 配置流程图

d. UJA1023 应用实例

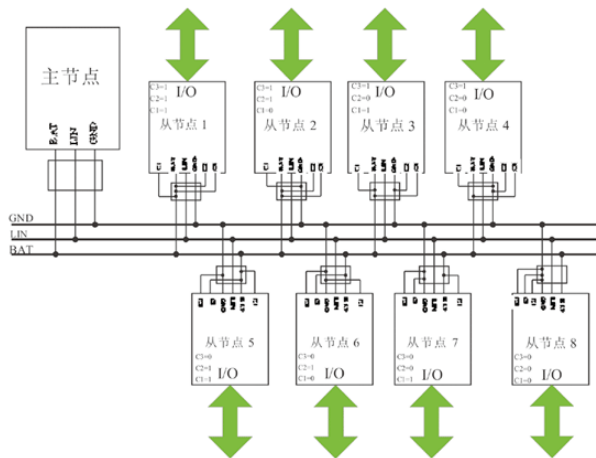


通用 I/O 口应用

由上图所示。单片 UJA1023 具有 8 个 I/O 口可作为通用 I/O 口使用。在图 4 中，P0、P1 用于读入开关量的值。P3 用于驱动发光二极管。P5 通过功率放大可驱动电机。P7 则对模拟量进行采集，P2 还可以驱动继电器，因而 UJA1023 在继电保护中也有“用武之地”。

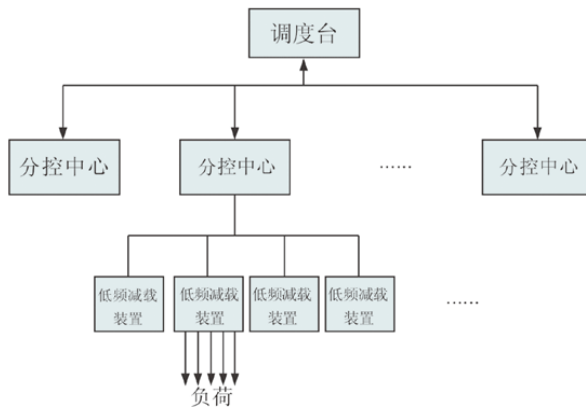
e. 多片 UJA1023 应用

使用即插方式，通过“单主多从”的方式，将多片 UJA1023 连成网络形式的应用。由下图所示为由 1 个主节点和 8 个从节点组成的基于单主多从模式的 LIN 网络。由图可知，每个从节点有 8 路 I/O，因此该网络总共有多达 64 路的控制口。可进行多种形式的多路控制。其特点就是应用灵活，配置方便，扩展性强，成本低廉，可靠性高。值得一提的是，下图只是一个子网络，如果需要更多的端口，完全可以按下图的形式进行多网络扩展，而网与网之间可以通过网关来连接。



基于单主多从模式的 LIN 网络

该形式十分适合于如下图所示的电力系统继电保护应用中。下位低频低压减载装置的设计目的是完成 35kV 系统根据电力系统频率和电压变化，按预定的控制方案切除电力负荷。



系统总体框图

上图中每台低频减载装置中的内部结构的设计思想，以图 5 为参考，一个装置中包含一个基于单主多从的子网络，即含有多片 UJA1023，而每片芯片的 I/O 口驱动若干的继电器，以在必要时控制继电器以切除负载。

由以上的分析不难得出结论，在工业自动化领域，对于数据的传输速率要求不高的传感器、数字量 I/O、显示和执行部件的控制连接固然可以使用 CAN，但是这样可以会造成浪费。在这种情况下，LIN 的成本廉价，抗干扰性强，使用单线连接的优越性就体现出来了。此外，在家用电器中也可使用 LIN 总线，例如在洗衣

总线接口技术

机中，微处理器通过 LIN 总线的传感器进行水位、水流、水温的测量，通过连在总线上的电动阀进行给水和排水控制。因此 LIN 总线技术的应用范围远不是局限于汽车工业，而在其他领域也同样有着不可忽视的广阔应用前景和现实意义。

并行总线

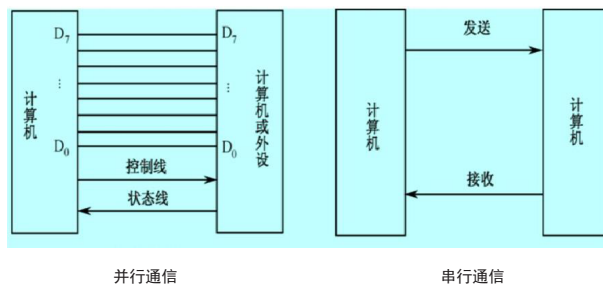
并行通讯：在计算机和终端之间数据传输是靠电缆或信道上的电流或电压变化实现的。如果一组数据的各数据位在多条线上同时被传输，这种传输方式称为并行通信。

并行通信有以下特点：

- ◆ 各数据位同时传输，传输速度快、效率高，多用在实时、快速的场合。
- ◆ 微机系统中最基本的信息交换方式。
- ◆ 并行传递的信息不要求固定的格式。
- ◆ 并行接口的数据传输率比串行接口快 8 倍，标准接口的数据传输率理论值为 1Mbps(兆比特/秒)。
- ◆ 并行传输的数据宽度可以是 1 ~ 128 位，甚至更宽，但是有多少数据位就需要多少根数据线，因此传输的成本较高。
- ◆ 并行通信抗干扰能力差。
- ◆ 在集成电路芯片的内部、同一插件板上各部件之间、同一机箱内个插件板之间的数据传输都是并行的。
- ◆ 以计算机的字长，通常是 8 位、16 位或 32 位为传输单位，一次传送一个字长的数据。
- ◆ 适合于外部设备与微机之间进行近距离、大量和快速的信息交换。
- ◆ 并行数据传输只适用于近距离的通信，通常传输距离小于 30 米。

并行通信时数据的各个位同时传送，可以字或字节为单位并行进行。并行通信速度快，但用的通信线多、成本高，故不宜进行远距离通信。计算机或 PLC 各种内部总线就是以并行方式传送数据的。并行和串行传

送方式如下图所示。



可以看出，在并行通信中，数据有多少位就需要有多少根传输线，而串行通信无论数据有多少位只需要一对传输线。因此，串行通信在远距离和多位数据传送时，有着明显的优越性。但并行通信的数据传输速度要优于串行通信。

(1) PCI 与 PCI Express

① PCI 总线的特性

PCI 总线是一种同步的独立于处理器的 32 位或 64 位局部总线，最高工作频率为 33MHz，峰值速度在 32 位时为 132MB/s，64 位时为 264MB/s，总线规范由 PCISIG 发布。ISA 总线相比，PCI 总线和有如下显著的特点：

◆ 高速性

PCI 局部总线以 33MHz 的时钟频率操作，采用 32 位数据总线，数据传输速率可高达 132MB/s，远超过以往各种总线。而早在 1995 年 6 月推出的 PCI 总线规范 2.1 已定义了 64 位、66MHz 的 PCI 总线标准。因此 PCI 总线完全可为未来的计算机提供更高的数据传送率。另外，PCI 总线的主设备 (Master) 可与微机内存直接交换数据，而不必经过微机 CPU 中转，也提高了数据传送的效率。

◆ 即插即用性

目前随着计算机技术的发展，微机中留给用户使用的硬件资源越来越少，也越来越含糊不清。在使用 ISA 板卡时，有两个问题需要解决：一是在同一台微机上使用多个不同厂家、不同型号的板卡时，板卡之间可能会有硬件资源上的冲突；二是板卡所占用的硬件资源可能会与系统硬件资源（如声卡、网卡等）相冲突。而 PCI 板卡的硬件资源则是由微机根据其各自的要

总线接口技术

求统一分配，决不会有任何的冲突问题。因此，作为 PCI 板卡的设计者，不必关心微机的哪些资源可用，哪些资源不可用，也不必关心板卡之间是否会有冲突。因此，即使不考虑 PCI 总线的高速性，单凭其即插即用性，就比 ISA 总线优越了许多。

◆ 可靠性

PCI 独立于处理器的结构，形成一种独特的中间缓冲器设计方式，将中央处理器子系统与外围设备分开。这样用户可以随意增添外围设备，以扩充电脑系统而不必担心在不同时钟频率下会导致性能的下降。与原先微机常用的 ISA 总线相比，PCI 总线增加了奇偶校验错 (PERR)、系统错 (SERR)、从设备结束 (STOP) 等控制信号及超时处理等可靠性措施，使数据传输的可靠性大为增加。

◆ 复杂性

PCI 总线强大的功能大大增加了硬件设计和软件开发的实现难度。硬件上要采用大容量、高速度的 CPLD 或 FPGA 芯片来实现 PCI 总线复杂的功能。软件上则要根据所用的操作系统，用软件工具编制支持即插即用功能酶设备驱动程序。

◆ 自动配置

PCI 总线规范规定 PCI 插卡可以自动配置。PCI 定义了 3 种地址空间：存储器空间，输入输出空间和配置空间，每个 PCI 设备中都有 256 字节的配置空间用来存放自动配置信息，当 PCI 插卡插入系统，BIOS 将根据读到的有关该卡的信息，结合系统的实际情况为插卡分配存储地址、中断和某些定时信息。

◆ 共享中断

PCI 总线是采用低电平有效方式，多个中断可以共享一条中断线，而 ISA 总线是边沿触发方式。

◆ 扩展性好

如果需要把许多设备连接到 PCI 总线上，而总线驱动能力不足时，可以采用多级 PCI 总线，这些总线上均可以并发工作，每个总线上均可挂接若干设备。因此 PCI 总线结构的扩展性是非常好的。由于 PCI 的设计是要辅助现有的扩展总线标准，因此与 ISA，EISA 及 MCA 总线完全兼容。

◆ 多路复用

在 PCI 总线中为了优化设计采用了地址线和数据线共用一组物理线路，即多路复用。PCI 接插件尺寸小，又采用了多路复用技术，减少了元件和管脚个数，提高了效率。

◆ 严格规范

PCI 总线对协议、时序、电气性能、机械性能等指标都有严格的规定，保证了 PCI 的可靠性和兼容性。由于 PCI 总线规范十分复杂，其接口的实现就有较高的技术难度。

② PCI 总线概述

PCI 总线是由 ISA (Industry Standard Architecture) 总线发展而来的，ISA 并行总线有 8 位和 16 位两种模式，时钟频率为 8MHz，总线带宽为： $8\text{bit} \times 8\text{MHz} = 64\text{Mbps} = 8\text{MB/s}$ 或 $16\text{bit} \times 8\text{MHz} = 128\text{Mbps} = 16\text{MB/s}$ 。在计算机出现初期的 386/486 时代，ISA 总线的带宽已经算是很大的了，应付 CPU 可以说是绰绰有余了。1987 年，IBM 公司推出 32 位 MCA (Micro Channel Architecture) 总线，总线带宽达到 40MB/s，迫于 IBM 的压力，Compaq、AST、Epson、HP、Olivetti 和 NEL 等 9 家公司联合于 1988 年 9 月推出 EISA (Extended ISA) 总线，EISA 总线仍然保持 ISA 总线的 8MHz 时钟频率，但将总线位宽提高到 32 位，总线带宽为： $32\text{bit} \times 8\text{MHz} = 256\text{Mbps} = 32\text{MB/s}$ ，并且与 ISA 总线完全兼容。

随着计算机技术的高速发展，CPU 的处理速度越来越快，EISA 总线的 32MB/s 带宽已经满足不了 CPU 的需求，CPU 外围总线带宽已经成为制约计算机处理能力继续提高的瓶颈。1991 年下半年，Intel 公司首先提出 PCI 总线的概念，并与 IBM、Compaq、AST、HP、DEC 等 100 多家公司成立 PCISIG 组织，联合推出 PCI 总线。

PCI 总线支持 32 位和 64 位两种位宽，时钟频率为 33MHz，总线带宽： $32\text{bit} \times 33\text{MHz} = 1056\text{Mbps} = 132\text{MB/s}$ 或 $64\text{bit} \times 33\text{MHz} = 2112\text{Mbps} = 264\text{MB/s}$ 。

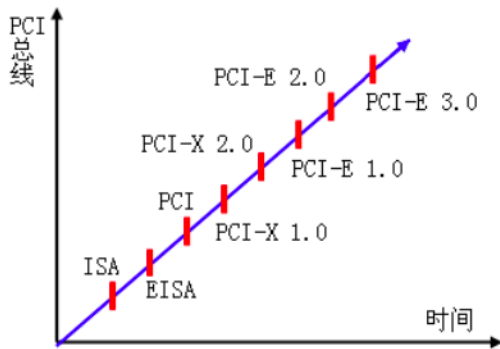
随着 PCI 总线的发展，PCI 总线又发展到 PCI-X 时代，与 PCI 总线相比，PCI-X 总线的位宽未改变，

总线接口技术

而是将时钟频率进行了提高。PCI-X 1.0 的时钟频率有 66MHz/100MHz/133MHz，总线带宽分别为：264MB/s, 400MB/s 和 532MB/s (32 位)，528MB/s, 800MB/s 和 1064MB/s (64 位)；PCI-X 2.0 的时钟频率有 266MHz/533MHz/1066MHz，总线带宽分别为：1064MB/s, 2132MB/s 和 4264MB/s (32 位)，2128MB/s, 4264MB/s 和 8512MB/s, PCI-X 与 PCI 总线在硬件结构上完全兼容。

与 PCI 总线相比，PCI-X 除掉在时钟频率高外，还在传输协议上进行了改进，PCI-X 采用了“寄存器到寄存器”的新协议，在新协议模式下，发送方发出的数据信号会被预先送入一个专门的寄存器内，并在寄存器中保存一个时钟周期，接收端只要在这个周期内做出响应，取走数据即可，而 PCI 协议中却没有这个缓冲，数据发出去后，如果对方因某种原因没能及时接收，数据将会被丢弃。

如今，PCI 总线已经发展到 PCI-Express 时代，这是一个串行高速总线，也是由 PCISIG 推出，分为 X1, X2, X4, X8, X12, X16 和 X32 七种模式，X1 模式有 2 对差分线，1 对收 1 对发，X2 模式有 4 对差分线，2 对收 2 对发，其它类推。PCI-E 1.0 的速率为 2.5Gbps，PCI-E 2.0 的速率为 5.0Gbps，PCI-E 3.0 的速率可达 8.0Gbps。PCI-E 正在替代着 PCI/PCI-X 的角色，相信在不久的将来，PCI/PCI-X 总线就如果 ISA/EISA 总线一样，进入历史的博物馆。

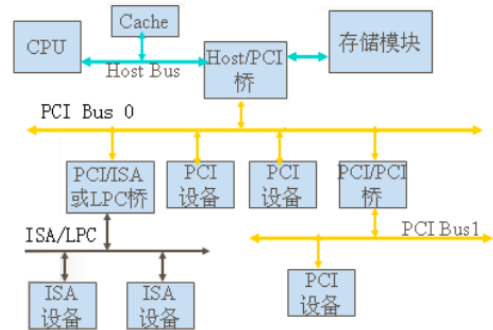


PCI 总线的发展示意图

③ PCI 总线的系统结构

PCI 总线是一种树型结构，并且独立于 CPU 总线，可以和 CPU 总线并行操作。PCI 总线上可以接 PCI 设备和 PCI 桥片，PCI 总线上只允许有一个 PCI 主设备，

其他的均为 PCI 从设备，而且读写操作只能在主从设备之间进行，从设备之间的数据交换需要通过主设备中转。



PCI 总线的系统结构

④ PCI 总线信号的定义及数据传输机制

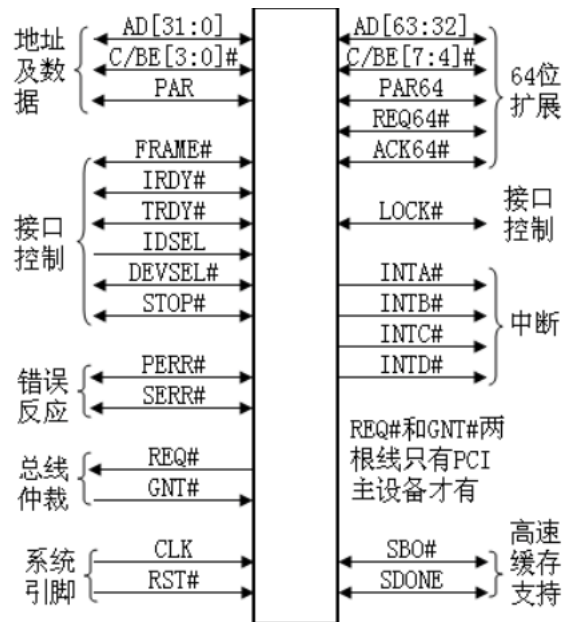
◆ PCI 总线信号的定义

首先说明两个概念：

主设备：在一个 PCI 应用程序中，取得了总线控制权的设备。

从设备：被主设备选中以进行通信的设备。

PCI 总线设备引脚如下图：



PCI 总线设备引脚

总线接口技术

PCI主设备最少需要49根线,从设备最少需要47根线,剩下的线可选。在介绍PCI管脚功能前,先来说明下PCI管脚信号的类型。

in: 输入信号 out: 输出信号

t/s: 双向三态信号 (Tri-state), 无效时为高组态

s/t/s: 持续三态信号 (Sustained Tri-state), 每次由且只由一个单元拥有并驱动的低有效双向、三态信号。驱动一个s/t/s信号到低的单元在释放该信号浮空之前必须要将它驱动到高电平至少一个周期。这个特点很重要,在后面我们分析PCI信号质量案例的时候会用到。

o/d: 漏极开路输出 (Open Drain)

#: 此符号代表该信号在低电平时有效

实际使用中需要上拉的信号有:

FRAME#, TRDY#, IRDY#, DEVSEL#, STOP#, PERR#, SERR#, LOCK#, REQ64#, ACK64#, REQ#, GNT#, AD[63:32], C/BE[7:4], PAR64等,上拉电阻一般为10kohm,未使用的PCI管脚也要处理,避免悬空。

不需要上拉的信号有:

AD[31:0], C/BE[3:0], PAR, IDSEL, CLK。

a. 系统引脚

CLK: in, 系统时钟,为所有PCI上的传输及总线仲裁提供时序。除RST#外,所有PCI信号都在CLK信号的上升沿采样。

RST#: in, 异步复位信号。

b. 地址及数据引脚

AD[31:0]: t/s, 地址数据复用引脚。FRAME#开始变为有效的那个时钟周期内AD[31:0]上传输的是地址。对于配置空间和存储空间,这是一个双字节地址,对于I/O空间,这个一个单字节地址。传输数据时,AD[7:0]为最低字节数据。

C/BE[3:0]#: t/s, 总线命令和字节允许复用引脚。在AD[31:0]上传输地址的时候,C[3:0]上传输的是总线命令,AD[31:0]上传输数据的时候,BE[3:0]#用作字

PAR: t/s, AD[31:0]和C/BE[3:0]#上的数据偶效验。PAR与AD[31:0]有相同的时序,但延迟一个时钟,在地址段后一个时钟,PAR稳定并有效;对于数据段,在写传输中,PAR在IRDT#有效一个时钟稳定并有效,而在读传输中,PAR在TRDY#有效后一个时钟周期稳定并有效。一旦PAR有效,它必须保持有效直到当前数据段完成后一个时钟。在地址段和写数据段,主PCI设备驱动PAR,在读数据段,目标从PCI设备驱动PAR。

c. 接口控制引脚

FRAME#: s/t/s, 帧开始信号。由当前总线主设备驱动,以说明一个操作的开始和延续。FRAME#有效,说明总线开始传输,当FRAME#维持有效时,说明总线传输继续进行,当FRAME#无效时,说明传送的最后一个字节正在进行。

IRDY#: s/t/s, 启动者准备好信号 (Initiator Ready)。说明传输的启动者完成当前数据传输的能力。在读操作中,IRDY#有效说明总线主设备已准备好接收数据。在写操作中,它说明AD[3:0]上已有有效数据。在IRDY#和TRDY#都有效的时钟周期完成数据传输。在IRDY#和TRDY#都有效之前,需要插入等待状态。

TRDY#: s/t/s, 目标设备准备就绪 (Target Ready)。说明传输的目标设备完成当前的数据传输能力。在写操作中,TRDY#有效说明目标设备已经准备好接收数据。在读操作中,它说明AD[31:0]上已有有效数据。

STOP#: s/t/s, 停止信号。说明当前的目标设备要求总线主设备停止当前传输。

LOCK#: s/t/s, 锁定信号。

IDSEL: in, 初始化设备选择 (Initialization Device Select)。在配置空间读写操作中,用作片选。

DEVSEL#: s/t/s, 设备选择。当驱动有效时,说明驱动它的设备已将其地址解码为当前操作的目标设备。

d. 仲裁引脚

REQ#: t/s, 申请。向仲裁器说明该单元想使用总线。这是一个点对点的信号,每个总线主设备都有自己的REQ#。

GNT#: t/s, 允许。仲裁器向申请单元说明其对总线

总线接口技术

的操作已被允许。这是一个点对点信号，每个总线主设备都有自己的 GNT#。

e. 错误反馈引脚

PERR#: s/t/s, 奇偶校验错误 (Parity Error)。该引脚用于反馈在除特殊周期外的其他传送过程中的数据奇偶校验错误。PERR# 维持三态，在检测到奇偶校验错误后，在数据结束后两个时钟周期，由接收数据的单元驱动 PERR# 有效。并至少持续一个时钟周期。只有发出 DEVSEL# 的单元才能发出 PERR#。

SERR#: o/d, 系统错误 (System Error)。用于反馈地址奇偶校验错误、特殊周期命令中的数据奇偶校验错误和将引起重大事故的其他灾难性的系统错误。

中断引脚

INTA#, INTB#, INTC#, INTD#: o/d, 中断输出。

f. 高速缓存支持引脚

一个能高速缓存的 PCI 存储器必须利用这两条高速缓存支持引脚作为输入，以支持写通 (write-through) 和回写 (write-back)。如果可高速缓存的存储器是位于 PCI 上，则连接回写高速缓存到 PCI 的桥路必须利用这两条引脚，且作为输出。连接写通高速缓存的桥路可以只使用一条引脚 SDONE。

SBO#: in/out, 监视补偿。当其有效时，说明对某条变化线的一次命中。当 SBO# 无效而 SDONE 有效时，说明了一次“干净”的监视结果。

SDONE: in/out, 监视进行。表明对当前操作的监视状态。当其无效时，说明监视结果仍未定。当有效时，说明监视已有结果。

g. 64 位总线扩充引脚

AD[63:32]: t/s, 地址数据复用引脚提供 32 个附加位。在一个地址段，传送 64 位地址的高 32 位。在数据段，传送 64 位中的高 32 位。

C/BE[7:4]#: t/s, 总线命令和字节允许复用引脚。

REQ64#: s/t/s, 请求 64 位传输。当其被当前总线主设备有效驱动时，说明总线主设备想作 64 位传输。

ACK#: s/t/s, 应答 64 位传送。在当前操作所寻址的

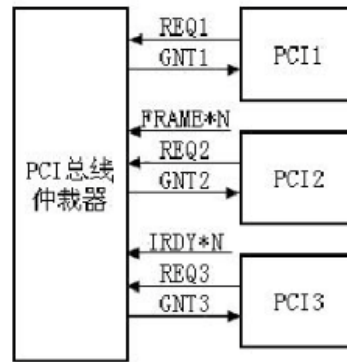
目标设备有效驱动该信号时，说明目标设备能够进行 64 位传输，ACK# 和 DEVSEL# 有相同的时序。

PAR64: t/s, 高双字偶校验。

◆ PCI 总线的传输机制

a. PCI 仲裁器工作原理

假设 PCI 仲裁器上接三个 PCI 设备，如下图所示。



假定 PCI1, PCI2, PCI3 三个设备的优先级相同，上电后，PCI 总线仲裁器会将 PCI1 设备的 GNT1 信号驱动为低电平 (有效)。即 PCI1 占用 PCI 总线。如果此时 PCI1 上没有数据传输，但 PCI2 上有数据要传输，PCI2 设备就会驱动 REQ2 为低电平来向仲裁器发出 PCI 总线使用请求。仲裁器在时钟的上升沿采样到 REQ2 信号，在下一个时钟沿，PCI2 将 FRAME 和 IRDY 驱动为低电平 (FRAME 和 IRDY 是所有 PCI 设备共用的)，仲裁器在时钟的上升沿采样到这两个信号后，就将 GNT2 驱动为低电平，应答 PCI2 的请求，这样，PCI2 就拥有了 PCI 总线的使用权。

这里有一个问题，为什么 PCI 总线仲裁器不在采样到 REQ2 信号有效后立即将 GNT2 置低，将 PCI 总线的使用权交给 PCI2 设备，而是要等到 PCI2 设备的 FRAME 和 IRDY 信号有效后才执行呢？

这主要是仲裁器避免某些 PCI 设备申请总线使用权而实际上不使用，所以要等到 FRAME 和 IRDY 信号有效后 (表示数据传输准备就绪) 才授权使用总线。如果仲裁器在采样到 REQ2 有效后，一直没有采样到 FRAME 和 IRDY 信号有效，仲裁器会不会一直等待下去呢？答案是不会，仲裁器内部有一个定时器，等到定时器超时后，仲裁器就不再等了。

总线接口技术

仲裁规则：优先级高设备抢占优先级低设备资源 假设三个 PCI 设备的优先级顺序为：PCI3>PCI2>PCI1。某个时刻，PCI1 正在使用 PCI 总线传输数据，如果此时 PCI2 想要使用 PCI 总线，于是将 REQ2 驱动为低，向 PCI 总线仲裁器申请 PCI 总线的使用权，此时 REQ1 和 REQ2 都处于有效状态，虽然 PCI2 的优先级高于 PCI1，但是总线仲裁器不会马上响应 PCI2 的请求。而是等待 PCI1 完成一次操作（注意：是一次操作，不是全部）后，PCI1 将 FRAME 和 IRDY 总线驱动为高（每完成一次操作都需要进行此操作的）。

PCI2 探测到 PCI 总线空闲后，将 FRAME 和 IRDY 信号驱动为低。总线仲裁器采样到这两个信号有效后，将 GNT1 驱动为高，同时将 GNT2 驱动为低，将总线的使用权转交给 PCI2。此时，如果 PCI1 完成了全部传输，会将 REQ1 驱动为高。如果 PCI1 还有数据等待传输，那么它会将 REQ1 一直保持为低电平，一直进行总线申请。一直等待到 PCI2 完成所有操作后，还会将 PCI 总线的使用权释放给 PCI1。

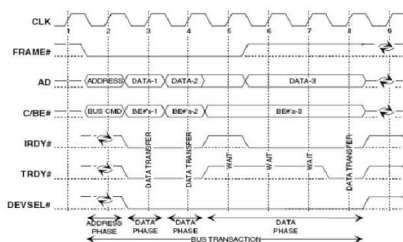
b. PCI 操作类型

面已经说过了，在 FRAME 信号有效后的第一时钟周期，AD[31:0] 总线上传输的是目标 PCI 器件的地址，C/BE[3:0] 上传输的操作命令类型。如下图所示。

C/BE[3:0]#	总线命令
0000	中断应答命令
0001	特殊周期命令
0010	I/O读命令
0011	I/O写命令
0100	
0101	
1000	保留命令
1001	
0110	存储器读命令
0111	存储器写命令
1010	配置读命令
1011	配置写命令
1100	存储器重复读命令
1101	双地址周期命令
1110	高速缓存线存储器读命令
1111	高速缓存线存储器写命令

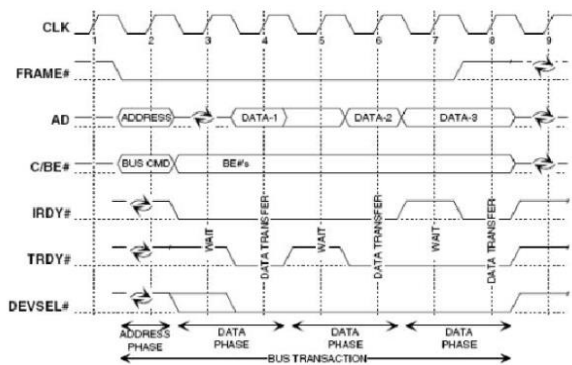
c. PCI 总线的读写时序

PCI 总线写时序如下图所示。



如上图所示，在 FRAME# 有效后的第一个时钟周期内，AD 上传输的是要写入目标 PCI 设备的地址信息，C/BE# 上传输的是命令类型（I/O 写命令为 0011），DEVSEL# 信号有效后，表明目标 PCI 设备已经被选择到，IRDY# 和 TRDY# 同时有效后，主 PCI 设备向目标 PCI 设备中传输要写入的数据，在第 5 个时钟周期时，IRDY# 和 TRDY# 同时变为无效状态，AD 总线上被插入一个等待周期，第 6 和第 7 个时钟周期时，IRDY# 有效，但是 TRDY# 无效，传输仍然不能有效进行，总线上被继续插入两个等待周期，第 8 个时钟周期时，IRDY# 和 TRDY# 都有效，数据传输继续。

PCI 总线的读时序如下图所示。



读操作同写操作类似，只是在 FRAME# 有效后的第一个时钟周期内，C/BE# 上传输的是读操作命令而已（I/O 读操作命令为 0010）。

⑤ PCI 总线接口设计

PCI 接口设计和 ISA 等 I/O 接口设计有着完全不同的设计思想，PCI 总线设备需要通过桥路控制器将 PCI 总线协议转换成本地总线协议，设计方案的实现大概有两种：

一种是选用可编辑逻辑器件 (CPLD)。使用 CPLD，用户可以灵活地开发出适合自己需要的具有特定功能的芯片，但 PCI 总线协议比较复杂，设计 PCI 控制接口难度较大，对于一般的工程项目来说，成本较大。现在有许多生产 CPLD 的厂商都提供经过严格测试的 PCI 接口功能模块，用户只需进行组合即可。另一种是选用 PCI 专用芯片组（又称桥接电路），通过专用芯片来实现完整的 PCI 主控模块和目标模块的功能，将复杂的 PCI 总线接口转换为相对简单的用户接口，用户只需设计转换后的总线接口。

总线接口技术

⑥ 基于 PCI 总线的系统结构

◆ 基于 PCI 总线的 DMA 高速数据传输系统

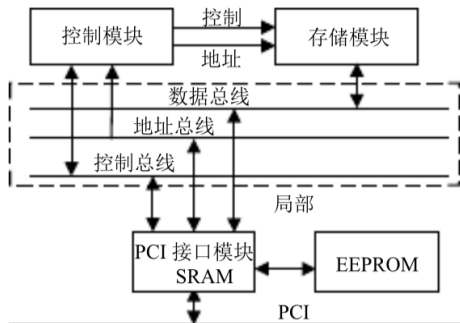
设计基于 PCI 总线接口技术的 DMA 传输系统，需要解决的主要问题是基于 PCI 总线的接口电路的设计，以及在 Windows 操作系统下的 DMA 驱动程序的开发。一般来说，一个 PCI 接口电路的设计主要围绕需要实现的功能来考虑。

a. PCI 总线的命令译码。接口电路对 C/BE[3:0] 和 IDSEL 总线信号进行译码，以确定来自主机的 PCI 命令是 I/O 访问还是存储器访问，并产生相应的控制信号。

b. 地址产生电路。在 PCI 突发传输方式中包含一个地址周期和若干个数据周期，因此 PCI 接口电路必须能向前端的数据采集电路提供连续的地址。

c. 控制信号产生。PCI 总线的数据传输基本上由 FRAME#、IRDY#、TRDY#、DEVSEL# 四个总线信号控制，因此，PCI 接口电路必须产生这些控制信号。

d. DMA 控制器的功能。在 DMA 传输时，接口电路必须能够实现 DMA。综合以上四个方面，设计中选用 PLX 公司的 PCI9054 作为接口芯片，能够完全实现 PCI 总线控制器以及 DMA 控制器的功能。基于 PCI 接口电路的数据传输系统原理框图如下图所示。系统中，存储模块用于采集数据的缓存；控制模块实现对存储模块的读写控制；EEPROM 存储 PCI 接口模块的配置信息。配置信息是 PC 机识别接口设备并分配相应资源的依据，如果配置不当，会引起计算机不能正常启动。在本文系统中，当主机启动时，EEPROM 完成对 PCI9054 内部的 PCI 总线配置寄存器和局部总线 (Local Bus) 配置寄存器的初始化。局部总线配置寄存器主要实现局部地址资源的分配，即局部总线端映射到 PCI 端的地址，包括存储器空间与 I/O 空间。



DMA 传输系统硬件框图

PCI9054 有 PCI BAR0 ~ PCI BAR5 六个基址寄存器 [5]。其中 PCI BAR0 和 PCI BAR1 用来访问 PCI9054 Local 端配置寄存器的基地址，BAR0 映射到内存，BAR1 映射 I/O 空间的基地址，就可以以内存方式或 I/O 的方式访问 PCI9054 Local 端配置寄存器。BAR2~BAR3 提供访问 Local 端的芯片，其他的保留不用。在本文系统设计中，BAR2 映射到 Local 端的“0 地址空间”，使用内存映射方式。当 Local 端内存基地址为 0x20000000，且大小为 1 M 时，配置 LAS0BA 为 0x20000000、LAS0RR 为 0xFF00000。

◆ DMA 数据传输系统中各个模块的设计

a. PCI 接口模块

PCI 接口模块由 PCI9054 来实现。PCI9054 是一个 32 位 33 MHz 总线主控 I/O 加速器，它是当今最先进的通用总线主控设备，能完美地实施 PCI 2.2 版规范的要求，可获得高达 132 MB/s 的突发传输速度。PCI9054 提供 PCI 总线、E2PROM 总线和 Local 总线三个接口。Local 总线有 J、M 和 C 三种工作模式。本文系统里将 Local 总线设置为 C 模式。

PCI9054 作为桥接芯片，在 PCI 总线和 Local 总线之间有 PCI Initiator (PCI 起始器)、PCI Target (目标数据传输模式) 和 DMA 传输模式三种数据传输模式。

PCI Initiator 传输模式支持局部总线直接访问 PCI 总线的存储空间和 I/O 空间，而 PCI Target 模式则支持 PCI 总线主设备通过 9054 访问局部总线的存储空间和 I/O 空间。当设定 PCI9054 为 PCI Target 模式时，支持三个 PCI to Local 地址空间。任何 PCI 总线的主设备可以以可编程的等待状态、总线宽度和突发传输功能访问 PCI9054 的三个 Local 空间。

PCI9054 有两个相互独立的 DMA 传输通道，通道 0 和通道 1 都支持从 PCI to Local 或 Local to PCI 的 DMA 数据传输。两个通道都具备块传输和分散 / 集中传输模式，通道 0 还支持请求模式的 DMA 传输。在 DMA 方式传输时，PCI9054 对于 PCI 和 Local 两总线来说都是主控设备。DMA 传输前，必须设置主控设备使能位 (PCICR[2]) 使 PCI9054 成为 PCI 总线的主设备。

分散 / 集中 DMA 传输模式要求主机在 PCI 空间或 Local 空间设定描述符模块，其中包括 PCI 和 Local 总线的起始地址、传输字节、传输方向和下一个描

总线接口技术

述符模块的地址。在 PCI9054 的 DMA 控制寄存器中，分散 / 集中模式使能位为 DMAMODE0[9] 或者 DMAMODE1[9]，发起传输的控制位为 DMACSR0 [1:0] 或者 DMACSR1[1:0]。PCI9054 在载入第一个描述符模块时发起传输，此后连续加载下一个模块，直至监测到链结束位 (DMADPR0[1] 或者 DMADPR1 [1]) 有效，则设置传输结束位 (DMACSR0[4] 或者 DMACSR1[4])，结束 DMA 传输。

DMA 块传输模式要求 PCI 主机或 Local 主机提供传输时的 PCI 空间和 Local 空间起始地址、传输字节、传输方向，并由主机设定 DMA 开始位启动一次数据传输，一旦传输完成，设置传输结束位，结束 DMA 传输。

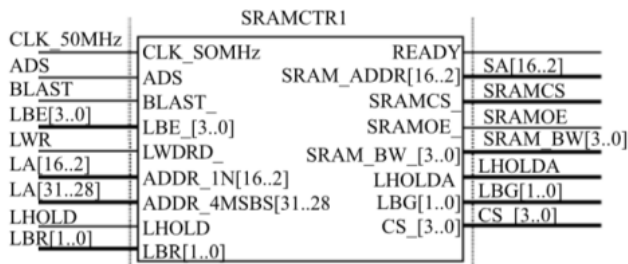
在雷达接收机综合测试系统中，运用 PCI Target 传输模式实现对 DMA 控制寄存器读写，运用 DMA 块传输模式实现对存储模块的数据传输。

b. 存储模块

利用 DMA 方式进行数据传输时，PCI9054 内部的 FIFO 大小远远不能满足数据传输的要求。因此在 Local 总线上外加存储器用于缓冲数据的存取。系统采用的存储芯片为 MT58L128L32F1，可以满足设计要求。

c. 控制模块

控制模块主要实现对存储模块的读写控制，其提供的控制信号如下图所示。

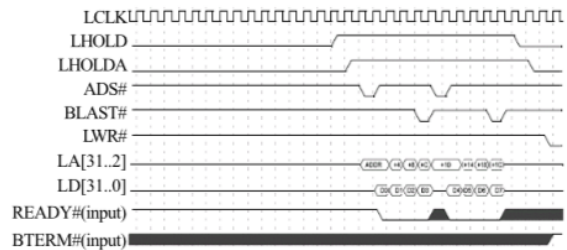


控制模块的接口

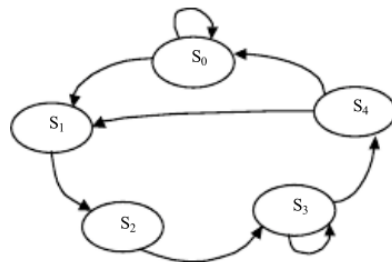
SRAM 需要的主要控制信号为：片选 (SRAMCS#)、使能输出 (SRAMOE#)、字节使能位 (SRAM_BW_ (3:0)) 及 SRAM 的地址。片选信号 (SRAMCS#) 由局部总线的高四位地址线 (LA31-LA28) 译码而得，SRAM 的地址信号 SA[16..2] 由局部总线输入地址信号 LA[16..2] 通过 CPLD 的内置计数器给出。低有

效的 READY 信号用于终止 PCI9054 的存储周期。控制模块根据相应的局部总线请求信号 LBR[1..0] 和 PCI9054 的总线请求信号 LHOLD 产生对应的应答信号 LGB[1..0] 和 LHOLDA。

设计时选用 ALTERA 的 CPLD EPM7128AE 芯片，用 VHDL 语言编写代码实现硬件控制逻辑的设计。PCI9054 产生 LHOLD 信号申请本地总线，如果本地总线空闲，EPM7128AE 向 PCI9054 产生 LHOLDA 应答信号，则 PCI9054 掌握本地总线控制权。然后产生 ADS# 信号，在下降沿处总线上开始出现要访问的地址，接着进行数据 burst 传输，每个 burst 读 4 个 32 b 数据，数据传送到最后一个字节时，PCI9054 产生 BLAST# 信号表示传输结束，则 EPM7128AE 将 READY# 信号置有效，等待读取第二组数据。时序关系图如下图所示。



PCI9054 DMA 传输时序



突发周期状态机设计

以上功能可以采用上图所示的状态机来描述。图中，S0 为空闲状态，S1 为突发开始状态，S2 为突发等待状态，S3 为突发重复状态，S4 为突发最后状态。用 VHDL 语言编写该状态机变化的代码，则可实现对存储模块的读写控制逻辑的硬件设计。

总线接口技术

(2) PXI 与 PXI Express

① PXI 总线概述

PXI 在 1997 年完成开发，并在 1998 年正式推出，它是为了满足日益增加的对复杂仪器系统的需求而推出的一种开放式工业标准。如今，PXI 标准由 PXI 系统联盟 (PXI Systems Alliance, PXISA) 所管理。该联盟由 20 多家公司组成，共同推广 PXI 标准，确保 PXI 的互换性，并维护 PXI 规范。

PXI 系统以工业标准 PCI 总线为基础，利用了丰富的 PCI 模版资源，PXI 继承了 PCI 的电器信号，使得 PXI 拥有如 PCI bus 的极高传输数据的能力，数据传输率可达 132MB / s 到 528MB/s，在软件上是完全兼容的。另一方面 PXI 采用和 compact PCI 一样的机械外形结构，因此也能同样享有高密度、坚固外壳及高性能连接器的特性。

PXI 这种新型模块化仪器系统是在 PCI 总线内核技术上增加了成熟的技术规范和要求形成的。它通过增加用于多板同步的触发总线和参考时钟、用于进行精确时的星形触发总线、以及用于相邻模块间高速通讯的局部总线来满足试验和测量用户的要求。PXI 规范在 CompactPCI 机械规范中增加了环境测试和主动冷却要求以保证多厂商产品的互操作性和系统的易集成性。PXI 将 Microsoft™ Windows NT™ 和 Microsoft™ Windows 95™ 定义为其标准软件框架，并要求所有的仪器模块都必须带有按 VISA 规范编写的 WIN32 设备驱动程序，使 PXI 成为一种系统级规范，保证系统的易于集成与使用，从而进一步降低最终用户的开发费用。

② PXI 的系统结构

◆ PXI 机械规范及其特性

由 CompactPCI 规范引入的 Eurocard 坚固封装形式和高性能的 IEC 连接器被应用于 PXI 所定义的机械规范，使 PXI 系统更适于在工业环境下使用，而且也易于进行系统集成。

a. 与 Compact PCI 共享的 PXI 机械特性

PXI 提供了两条与 Compact PCI 标准兼容的途径：

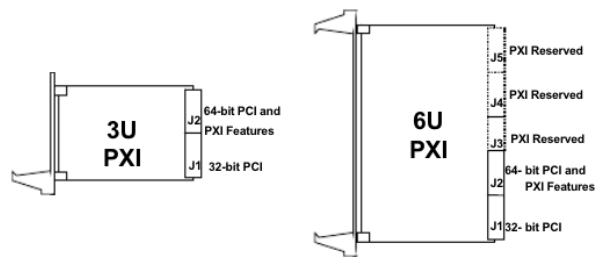
- 高性能 IEC 连接器

PXI 应用了与 Compact PCI 相同的、一直被用在象远距离通讯等高性能领域的高级针 - 座连接器系统。这种由 IEC-1076 标准定义的高密度 (2mm 间距) 阻抗匹配连接器可以在各种条件下提供尽可能好的电气性能。

- Eurocard 机械封装与模块尺寸

PXI 和 Compact PCI 的结构形状完全采用了 ANSI 310-C、IEC-297 和 IEEE 1101.1 等在工业环境下具有很长应用历史的 Eurocard 规范。这些规范支持小尺寸 (3U=100 mm x 160 mm) 和大尺寸 (6U=233.35 mm x 160 mm) 两种结构尺寸。IEEE1101.10 和 IEEE1101.11 等最新的 Eurocard 规范中所增加的电磁兼容性 (EMC)、用户可定义的关键机械要素、以及其它有关封装的条款均被移植到 PXI 规范中。这些电子封装标准所定义的坚固而紧凑的系统特性使 PXI 产品可以安装在堆叠式标准机柜上，并保证在恶劣工业环境中应用时的可靠性。

下图所示的是 PXI 仪器模块的两种主要结构尺寸及其接口连接器，其中，J1 连接器上定义了标准的 32-bit PCI 总线，所有的 PXI 总线性能定义在 J2 连接器上。PXI 机箱背板上包括可连接 J1 和 J2 连接器的所有 PXI 性能总线，对仪器模块来讲，这些总线可以有选择地使用。



下图是用于说明一个完整 PXI 系统的基本组成部分。PXI 规定系统槽 (相当于 VXI 的零槽) 位于总线的最左端，而 Compact PCI 系统槽则可位于背板总线的任何地方。PXI 规范定义唯一确定的系统槽位置是为了简化系统集成，并增加来自不同厂商的机箱与主控机之间的互操作性。PXI 还规定主控机只能向左扩展其自身的扩展槽，不能向右扩展而占用仪器模块插槽。

总线接口技术

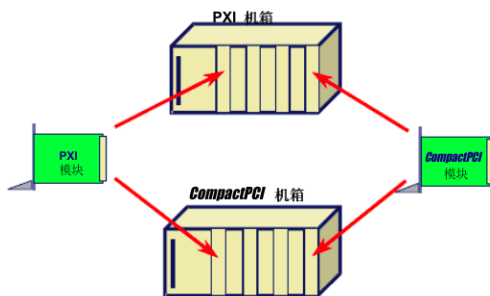


b. 新增加的电气封装规范

除了将 Compact PCI 规范中的所有机械规范直接移植进 PXI 规范之外，为了简化系统集成，PXI 还增加了一些 Compact PCI 所没有的要求。如前所述，PXI 机箱中的系统槽必须位于最左端，而且主控机只能向左扩展以避免占用仪器模块插槽。PXI 还规定模块所要求的强制冷却气流流向必须由模块底部向顶部流动。PXI 规范建议的环境测试包括对所有模块进行温度、湿度、振动和冲击试验，并以书面形式提供试验结果。同时，PXI 规范还规定了所有模块的工作和存储温度范围。

c. 与 Compact PC 的互操作性

如下图所示，PXI 的重要特性之一是维护了与标准 Compact PCI 产品的互操作性。但许多 PXI 兼容系统所需要的组件也许并不需要完整的 PXI 总线特征。例如，用户或许要在 PXI 机箱中使用一个标准 Compact PCI 网络接口模块，或者要在标准 Compact PCI 机箱中使用 PXI 兼容模块。在这些情况下，用户所需要的是模块的基本功能而不是完整的 PXI 特性。



◆ PXI 规范的电气性能

	参考时钟	触发线	星形总线	局部总线
VXI	10 MHz ECL	8 TTL & 2 ECL	仅 D 尺寸系统	12 线
PXI	10 MHz TTL	8 TTL	每槽一根	13 线

PXI 与 VXI 总线面向仪器领域的扩展性能比较

许多仪器应用场合需要而 ISA 总线、PCI 总线或 CompactPCI 背板总线所没有的系统定时能力，PXI 总线通过增加专门的系统参考时钟、触发总线、星形触发线和模块间的局部总线来满足高精度定时、同步与数据通信要求。PXI 不仅在保持 PCI 总线所有优点的前提下增加了这些仪器特性，而且可以比台式 PCI 计算机多提供三个仪器插槽，使单个 PXI 总线机箱的仪器模块插槽总数达到 7 个。PCI 总线与 VXI 总线面向仪器领域的扩展性能比较参见表 2-7。

a. 参考时钟

PXI 规范定义了将 10MHz 参考时钟分布到系统中所有模块的方法。该参考时钟可被用作同一测量或控制系统中的多卡同步信号。由于 PXI 严格定义了背板总线上的参考时钟，而且参考时钟所具有的低时延性能使各个触发总线信号的时钟边缘更适于满足复杂的触发协议。

b. 触发总线

如上表所示，PXI 不仅将 ECL 参考时钟改为 TTL 参考时钟，而且只定义了 8 根 TTL 触发线，不再定义 ECL 逻辑信号。这是因为保留 ECL 逻辑电平需要机箱提供额外的电源种类，从而显著增加 PXI 的整体成本，有悖于 PXI 作为 21 世纪主流测试平台的初衷。

使用触发总线的方式可以是多种多样的。例如，通过触发线可以同步几个不同 PXI 模块上的同一种操作，或者通过一个 PXI 模块可以控制同一系统中其它模块上一系列动作的时间顺序。为了准确地响应正在被监控的外部异步事件，可以将触发从一个模块传给另一个模块。一个特定应用所需要传递的触发数量是随事件的数量与复杂程度而变化的。

c. 星形触发

PXI 星形触发总线为 PXI 用户提供了只有 VXI D 尺寸

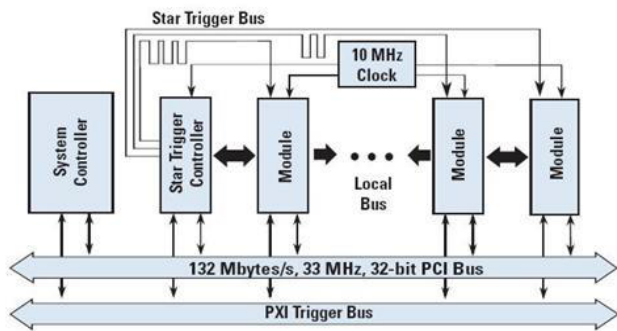
总线接口技术

= 系统才具有的超高性能 (Ultra-high performance) 同步能力。如图 2-31 所示, 星形触发总线是在紧邻系统槽的第一个仪器模块槽与其它六个仪器槽之间各配置了一根唯一确定的触发线形成的。在星形触发专用槽中插入一块星形触发控制模块, 就可以给其它仪器模块提供非常精确的触发信号。当然, 如果系统不需要这种超高精度的触发, 也可以在该槽中安装别的仪器模块。

应当提出, 当需要向触发控制器报告其它槽的状态或报告其它槽对触发控制信号的响应情况时, 就得使用星形触发方式。PXI 系统的星形触发体系具有两个独特的优点: 一是保证系统中的每个模块有一根唯一确定的触发线, 这在较大的系统中, 可以消除在一根触发线上组合多个模块功能这样的要求, 或者人为地限制触发时间。二是每个模块槽中的单个触发点所具有的低时延连接性能, 保证了系统中每个模块间非常精确的触发关系。

d. 局部总线

如下图所示, PXI 局部总线是每个仪器模块插槽与左右邻槽相连的链状总线。该局部总线具有 13 线的数据宽度, 可用于在模块之间传递模拟信号, 也可以进行高速边带通讯而不影响 PCI 总线的带宽。局部总线信号的分布范围包括从高速 TTL 信号到高达 42V 的模拟信号。



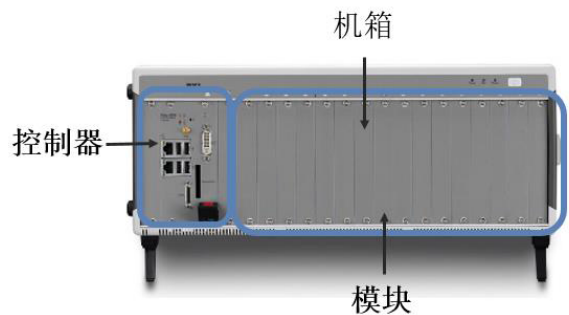
e. PCI 性能

除了 PXI 系统具有多达八个扩展槽 (一个系统槽和七个仪器模块槽), 而绝大多数台式 PCI 系统仅有三个或四个 PCI 扩展槽这点差别之外, PXI 总线与台式 PCI 规范具有完全相同的 PCI 性能。而且, 利用 PCI-PCI 桥技术扩展多台 PXI 系统, 可以使扩展槽的数量理论上最多能扩展到 256 个。其它的 PCI 性能还包括:

- 33MHz 性能
- 32-bit 和 64-bit 数据宽度
- 132MB/s(32-bit) 和 264MB/s(64-bit) 的峰值数据吞吐率
- 通过 PCI-PCI 桥技术进行系统扩展
- 即插即用功能

③ PXI 系统的组成

PXI 系统由三个基本部分组成 —— 机箱、系统控制器和外围模块。



简仪科技的标准的 18 槽 PXI 机箱中, 包括一个嵌入式系统控制器和 17 个外围模块

PXI 机箱

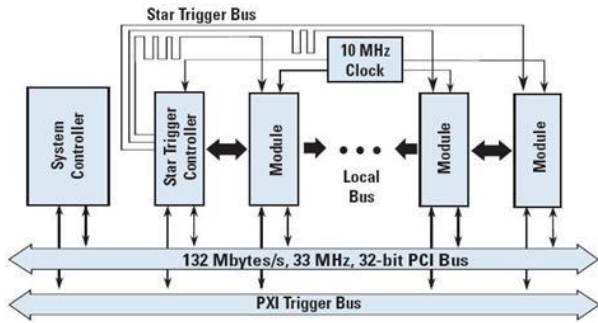
PXI 机箱为系统提供了坚固的模块化封装, 通常为 4 槽、6 槽、8 槽、14 槽或 18 槽的 3U 或 6U 机箱。U(rack unit) 是一种测量单位, 用来描述安装在 19 或 23 英寸 (指宽度) 机架上的设备的高度。1U 等于 44.45mm (1.75 英寸)。装在机架上的设备的尺寸大小通常用 U 来描述。专用机箱还可在交流电源和直流电源中选择, 以及是否集成信号调理功能。很多 PXI Express 机箱中都可以容纳 PXI 和 PXI Express 外围设备, 而有些具有混合插槽的机箱, 还可以容纳 PXI Express 外围设备或者兼容混合插槽的 PXI 设备。利用这些机箱可以配置多种 PXI 系统从而满足应用需求。



简仪科技的 PXI 机箱提供从 6 槽到 18 槽机箱等多种选择, 部分机型还带有集成的外设 (如 LCD 显示等)

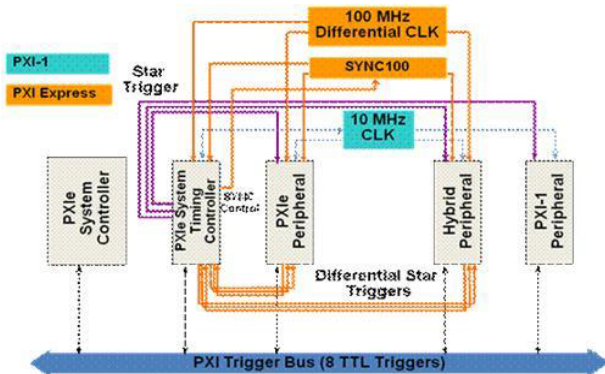
总线接口技术

机箱中具有高性能的 PXI 背板，该背板包括 PCI 总线，定时总线以及触发总线。PXI 模块化仪器系统中增加了专用的 10MHz 系统参考时钟、PXI 触发总线、星形触发总线和槽与槽之间的局部总线，从而在保持 PCI 总线所有优势的同时，满足高级定时、同步和相邻槽直接通信等应用中的需求。



PXI 的定时与触发总线 - PXI 在其背板上，将工业标准的 PC 机组件（如 PCI 总线）与高级触发和同步扩展组合在一起

以 PXI 的功能为基础，PXI Express 中提供了更多的定时和同步功能——100MHz 的差分系统时钟、差分信号传输以及差分星形触发总线。采用差分时钟和同步，PXI Express 系统中仪器时钟的抗噪声性能进一步提高，并且可以以更高速率传输数据。



PXI Express 的定时与触发总线扩展了 PXI 总线的定时与触发总线功能，并在背板中增加了差分系统时钟、差分信号传输和差分星形触发总线。

PXI 控制器

大多 PXI 机箱在最左端的插槽（插槽 1）中包含一个系统控制器插槽。在为用户的应用选择最佳系统控制器时，有几种选择可供参考，包括以台式机、工作站、服务器或笔记本电脑作为外部控制器，或者选择装有 Microsoft 操作系统（Windows 7/Vista/XP）高性能嵌入

式控制器。外部控制器有两种可选类型，分别为通过笔记本电脑控制 PXI 或通过 PC 机控制 PXI。

PXI 嵌入式控制器

嵌入式控制器无需使用外部 PC 机，从而可以使 PXI 机箱成为一个完整系统。这些嵌入式控制器具有一些标准特性，如集成有 CPU、硬盘驱动器、RAM、以太网口、视频输出、键盘/鼠标接口、串口、USB 和其他一些外围设备，以及已安装好的 Microsoft Windows 和所有设备驱动。基于 PXI 或者 PXI Express 的系统中都包含了以上这些特性；用户也可以自己选择操作系统，包括 Windows 7/Vista/XP 或者 LabVIEW Real-Time。

PXI 嵌入式控制器一般采用标准的 PC 机组件，封装在一个小巧的 PXI 封装中。例如，简仪科技的 PXIe-3985 控制器中包含一个 2.4GHz Intel Core i7-4700E 四核处理器（单核最大 3.4 GHz，采用英特尔睿频加速技术）嵌入式控制器，可选硬盘驱动或者固态驱动器，还具有两个千兆位以太网端口和标准的 PC 外围设备，如高速 USB、ExpressCard/34、串口与并口等等。



简仪科技的 PXIe-3985 控制器中包含一个 2.4GHz Intel Core i7-4700E 四核处理器（单核最大 3.4 GHz，采用英特尔睿频加速技术）嵌入式控制器

对于便携式系统以及希望易于将系统从一个地方搬移到另一个地方的“单箱”应用来说，嵌入式控制器是一个理想的选择。

通过笔记本电脑控制 PXI

采用 ExpressCard MXI (Measurement eXtensions for Instrumentation，面向仪器系统的测量扩展) 和 PCMCIA CardBus 接口套件，用户可以直接在笔记本电脑上通过软件透明的链路连接来控制 PXI 系统。开机时，笔记本电脑会默认 PXI 系统中的所有外围模块都是 PCI 板卡，之后用户就可以通过笔记本电脑来控制

总线接口技术

制这些设备。通过笔记本电脑控制 PXI，需要在笔记本上插入一块 ExpressCard/PCMCIA CardBus 卡，并在 PXI 机箱第 1 槽中插入一个 PXI/PXI Express 连接模块，两者之间用铜芯电缆相连。



ExpressCard MXI 接口套件



PCMCIA CardBus 接口套件

通过笔记本电脑控制 PXI

ExpressCard MXI 接口套件为笔记本电脑和 PXI 机箱间提供了一个 110MB/s 的 PCI Express 至 PCI 的桥接。PCMCIA CardBus 接口套件则为笔记本电脑和 PXI 机箱间提供了一个 50 MB/s 的 PCI 至 PCI 桥接。这样，通过笔记本电脑控制 PXI，可以实现移动的 / 便携的 PXI 系统。任何与 ExpressCard MXI 或 PCMCIA CardBus 兼容的笔记本电脑都可以用来远程控制 PXI 系统。

通过 PC 机控制 PXI

采用 MXI-Express 和 MXI-4 接口套件，用户可以直接从台式机、工作站或服务器电脑上控制 PXI 系统。用户可以在 PC 机上通过软件透明和驱动透明的连接链路来控制 PXI 系统。开机时，电脑系统会默认 PXI 系统中的所有外围模块都是 PCI 板卡，所以用户可以通过控制器来使用这些设备。通过 PC 机控制 PXI，需要在电脑中插入一块 PCI/PCI Express 板卡，并在 PXI 机箱第 1 槽中插入一个 PXI/PXI Express 模块，两者之间用铜芯电缆或光纤电缆相连。



采用双端口 MXI-Express 的远程控制可以同时控制两个 PXI 机箱，总吞吐率为 528MB/s

MXI-Express 接口套件为 PC 机和 PXI 机箱间提供了一个 132MB/s 的 PCI Express 至 PCI 的桥接。采用简仪的 PCIe-PXIe-68638 接口套件，可以通过同一台 PC 机控制 PXI 系统。用户还可以利用 MXI-Express 接口，通过 3m 或 7m 的铜芯电缆来配置星形拓扑结构的多机箱 PCI 系统。这种接口与 x1、x4、x8 和 x16 的 PCI Express 插槽兼容，因此该可选方案适用于大部分的台式电脑。



采用 MXI-4 的远程控制，可以实现通过 PC 机控制 PXI 以及多机箱的 PXI 系统

MXI-4 接口套件为 PC 机和 PXI 系统间提供了一个 78 MB/s 的 PCI 至 PCI 桥接。MXI-4 接口套件中有低成本铜芯电缆链接选择，也有针对远距离应用和电气隔离应用的光纤链接选择。用户还可以采用 MXI-4 创建多机箱的 PXI 系统。采用 MXI-4 链接，用户可以实现菊花链式或星形拓扑结构的多机箱系统。

采用 PXI 外部控制器，用户可以通过台式电脑或笔记本电脑来远程控制 PXI 系统，从而以最低的成本获得最优的处理器性能。由于所有的外部控制产品都是软件透明的，所以不需要额外的编程。用户可以购买多台台式电脑、工作站或服务器电脑，然后使用 MXI-Express 或者铜芯 / 光纤电缆的 MXI-4 串行链接，来远程控制 PXI 系统。

Rack-Mount 控制器

提供外置的 1U rack-mount 控制器，可用于复杂计算或大数据量存储等应用。这些控制器可通过 MXI-Express 或 MXI-4 连接 PXI 或 PXI Express 机箱，此时机箱中的外围模块看上去就像 PC 中的本地模块一样。



Rack-mount 控制器通过 MXI-Express 或 MXI-4 可连接 PXI 或 PXI Express 机箱

总线接口技术

◆ PXI 外围设备

PXI 是一种开放的工业标准，目前共有来自 70 多家设备商的 1500 多种模块，覆盖大部分 PXI 应用。

- 模拟输入与输出
- 边界扫描
- 总线接口与通信
- 载波发生
- 数字输入与输出
- 数字信号处理
- 功能测试与诊断
- 图像采集
- 运动控制
- 可编程电源
- 开关模块
- 定时输入与输出模块
- RF 与通信模块

PXI 为各种应用——从高分辨率的直流应用到 6GHz 的射频应用——提供了具有业界最高带宽和最低延时的总线和模块化 I/O。因为 PXI 大多都与 CompactPCI 兼容，所以用户可以在 PXI 或者 PXI Express 系统中使用 3U 或 6U 尺寸的 CompactPCI 模块。另外，用户还可以使用转接模块在 PXI 系统中安装 CardBus/PCMCIA 和 PMC (PCI Mezzanine Card, PCI 夹层卡) 板卡。例如，利用 PXI-8221 PC Card 转接模块，用户可以将 CardBus 和 PCMCIA 设备接入 PXI 系统。简仪公司提供了多种 PXI 模块，可用于仪器系统、数据采集、开关切换、运动控制、机器视觉、高级同步以及与其它总线的接口，基于这些模块用户可以构建出各种各样的 PXI 系统。

PXI 系统还支持标准的用于多种总线通信的软件和硬件，从而使用户对于独立仪器或 VXI 系统的既有投资不致浪费。例如，使用 PXI-GPIB 模块将基于 GPIB 的仪器连接到 PXI 系统中与使用 PCI-GPIB 卡直接进行仪器控制没有任何区别。这两种方法所用的是同一种软件。此外，多种 PXI 与 VXI 的接口方法可供选择。

◆ 软件架构

基于 Windows 的 PXI 系统的开发与操作与标准的基于 Windows 的 PC 机系统的开发与操作差异性很小。另外，因为 PXI 背板使用的是工业标准的 PCI 总线，所以大多情况下，对 PXI 模块的软件编程也与 PCI 板卡一样。因此，将基于 PC 系统的程序移植到 PXI 系统时，无需重写现有的应用程序和示例代码。

PXI Express 系统同样也具有软件方面的兼容性，从而可以保护用户在软件方面的既有投资。由于 PCI

Express 使用了与 PCI 相同的驱动和操作系统模型，这些规范就保证了与基于 PCI 系统的软件兼容性。因此，设备商和客户都不需要改变驱动或应用软件来适应基于 PCI Express 的系统。

举例说明，由于 PXI 和 PXI Express 系统可以使用与 PCI 相同的驱动，因此控制简仪 PXI-62005 多功能数据采集模块的软件与控制 PC 机中 PCI-62005 板卡的软件是完全相同的。



两种不同的封装使用同一软件标准。
就软件而言，对 PXI 模块 (下方) 的编程和对 PCI 板卡 (上方) 的编程是一样的

对于要求确定性循环速率和无头操作 (即没有键盘、鼠标或显示器) 的、具有严格时间确定性要求的应用来说，可以使用实时软件架构替代基于 Windows 的系统。实时操作系统可以帮助你按照优先级顺序对任务进行排序，从而使处理器在必要时总会优先处理最重要的任务。这样，应用程序的执行结果就完全可以预测，而且定时确定性更高。

PXI 规范中提出了基于 Microsoft Windows 操作系统的 PXI 系统软件框架。因此，控制器可以使用工业标准的应用程序接口，如 Visual studio, LabVIEW、Visual Basic、Visual C/C++ 等等。PXI 组件需要初始化文件以定义系统配置和系统功能。最后，采用虚拟仪器软件架构 (Virtual Instrument Software Architecture, 即已被仪器领域广泛采纳的 VISA)，可实现对 VXI、GPIB、串口和 PXI 仪器的配置和控制。

虚拟仪器 3.0 的驱动设计

驱动的必要性和必要性

驱动程序一般是指安装在某种操作系统上，供上层应用程序调用以操作硬件的一组函数。驱动程序提供面向应用程序的 API 接口函数，把应用程序对 API 函数的调用翻译成硬件可以理解的底层寄存器操作。驱动程序封装了具体的硬件操作，使复杂的硬件操作对于软件开发者透明化，这样应用软件开发人员不必了解具体的硬件操作细节就能开发面向各种应用的程序。

大多数操作系统下的设备驱动程序可以分为用户层的“用户驱动”和内核层的“设备驱动”，它们之间通过操作系统提供的 API 接口函数进行交互。

“设备驱动”运行于操作系统的内核层，执行设备的添加与删除、资源申请与释放以及其他具体的硬件操作，例如中断响应、直接存储器访问 (DMA) 等，通常这些操作不能在用户层进行。“设备驱动”主要由“用户驱动”研发团队使用，基本要求是：安全可靠、可维护性强、可扩展性高与开发效率高。

“用户驱动”进一步对设备的功能抽象化，由应用程序开发者使用，为应用程序开发者提供尽可能透明的接口 API 函数。Windows 操作系统下的 Windows 驱动模型 (WDM) 就是这种结构。其基本要求是：安全可靠、友好易用、易于扩展。



(1) 虚拟仪器驱动程序的定义

仪器驱动是一套可被用户调用的子程序库，利用它调用相应的一些函数就可以完成对仪器各种功能的操作。

针对仪器的驱动最早成型于 20 世纪 70 年代，主要目

的是程控具备 GPIB 控制接口的台式仪器，用户层编程语言以 BASIC 为代表。80 年代开始出现了插卡式仪器，用户层编程语言以 C 为代表，90 年代出现了 SCPI(程控仪器标准命令)，出现了面向模块化仪器驱动程序。

虚拟仪器驱动的基本目标是友好地实现软件、硬件的无缝连接和方便、可靠地应用。

(2) 虚拟仪器驱动程序的发展历史

诞生于 1987 年的 VXI (VME eXtensions for Instrumentation) 总线技术正式开启了开放总线技术的模块化仪器时代，并推动了仪器总线标准、仪器驱动技术的规范化，目前 VXI 总线技术主要用在航空电子等军用自动化测试系统中，在其他更广阔的领域逐渐被更为流行的 PXI(e) 总线技术所取代。VXI 总线主要由 VXIbus Consortium(VXI 总线联盟) 推动，成立于 1993 年的 VXIplug&play Systems Alliance (VXIplug&play 联盟) 来制定 VXI 总线硬件、GPIB 总线接口仪器及其软件的互联互通标准，即制定 VXIplug&play (插拔使用) 规范，简称 VPP 规范。VPP 联盟目标是提高 VXI 技术最终用户的使用简易性。

IVI Foundation (IVI 基金会) 1998 年成立，IVI 基金会推出新一代仪器驱动技术规范 IVI (Interchangeable Virtual Instruments——可互换式虚拟仪器)，替换 VXIplug&play 标准。IVI 致力于实现仪器驱动器的可互换性、开发灵活性、测试高效性以及保证测试品质。目前的 IVI 驱动按照应用结构分为三种：IVI-COM, IVI-C 以及 IVI.NET。2002 年，VXIplug&play Systems Alliance 并入 IVI Foundation。VPP 规范继续存在，并且 VPP 规范对仪器驱动程序的要求不仅适用于 VXI 仪器，也同样适用于 GPIB 仪器、PXI(e) 仪器、串行口仪器、网络仪器、USB 仪器等。

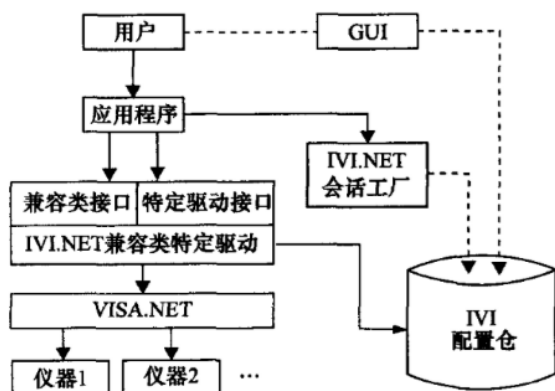
VISA (Virtual Instrument Software Architecture, 虚拟仪器软件构架) 属于 VPP 规范中的 VPP-4 系列，最早由 VXIplug&play Systems Alliance 提出，现在由 IVI Foundation 继续推动，是 IVI 规范的一部分，VISA 是一个位于底层“设备驱动”与上层“用户驱动”之间的软件层。VISA 为虚拟仪器提供了标准化的 I/O 接口软件规范，虚拟仪器软件结构中的标准 I/O 接口软件通常称为 VISA 库。

IVI.NET 是继 IVI-C 和 IVI-COM 之后的最新一代 IVI 驱动标准，它继承了 .NET 的诸多优点并拓展了传统 IVI

虚拟仪器 3.0 的驱动设计

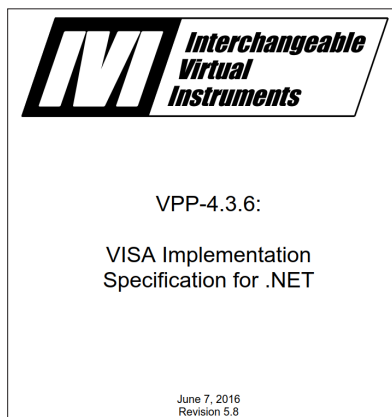
的功能。 .NET Framework 是微软为实现异质语言和应用平台高度交互性，基于开放互联网标准和协议构建的新一代计算和通信平台， .NET 技术具有如下突出优点：

- ① 多语言，跨平台支持。 .NET Framework 可以个多种语言，例如 C#、 C++、 C、 VB 等等；
- ② 网络化环境下的编程更加简单；
- ③ 数据类型更加丰富， API 更加简单；
- ④ 结构清晰，易学易用；
- ⑤ 错误处理机制更加完善。



IVI 驱动应用模型如上

IVI.NET 在 IO 驱动层采用 VISA.NET 驱动规范与各个仪器硬件进行通信。开发 IVI.NET 仪器驱动实际上就是利用 .NET 组间接口开发具有 IVI 规范的仪器驱动，可以直接使用 Visual Studio 进行开发，最新规范更新到 2016 年。



基于 .NET 虚拟仪器驱动规范

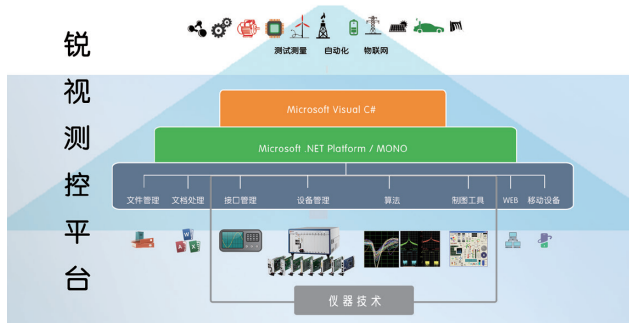
锐视测控平台的驱动设计

尽管虚拟仪器驱动在规范层面已经引入了 .NET 技术，但是目前的发展是差强人意的，原因大致如下：

- ① .NET 技术是微软主导的通用技术，在仪器行业没有公司、组织进行持续的、实质性的专注研发投入，目前的很多应用是自发的、分散的；
- ② 没有系统级别的策略和技术支持与投入；
- ③ 没有开发和使用的生态系统；

以 IVI 系列规范为例，IVI-C 驱动广泛的被 C 程序员使用，IVI-COM 驱动用于微软的组件，IVI.Net 在 2010 被提出，但开发的很少，到 2017 年还未成为虚拟仪器领域的主流。特别是 IVI .NET 对于 PXI(e) 总线类的虚拟仪器缺少专门的 PXI 设计驱动架构。

2016 年一家中国公司正式提出了锐视测控平台，该平台具有强大、易用、开源的特点。借助于 Microsoft® .NET 平台和 Microsoft® Visual Studio 开发环境的强大技术支撑，锐视测控平台成为业界第一个功能强大、且完全开源的测控系统专业开发平台。锐视测控平台提供基于 Visual C# 语言和 x86 架构的一系列软硬件解决方案，使得仅具有基本 C 语言基础的测控工程师、学生也可以轻松快速地开发出功能强大、界面专业、易于维护和扩展的测控系统，从而大大提高测控系统的开发效率，实现软硬件之间的无缝连接。



锐视测控平台技术架构

锐视测控平台是基于 Microsoft® .NET 平台和 Microsoft® Visual Studio 开发环境的测控系统开发平台。在锐视测控平台下，开发者不但可以获得大量基于 Microsoft® Visual Studio 开发环境的成熟技术和强大功能，例如图形界面、文件 I/O、算法类库、硬件驱动、仪器接口、网络访问等，还可以第一时间获得更

虚拟仪器 3.0 的驱动设计

高效的运行引擎更新，功能更强大的基础开发环境，以及最新的技术和设备支持。同时，借助 Microsoft® .NET 的跨平台支持，开发者还可以很容易地将测控系统移植到 Microsoft® Windows™ 之外的系统平台，例如 Linux、Mac OS X、iOS 和 Andriod 等。

为了提高测控工程师在 .NET 平台下的系统开发效率，降低测控系统开发入门门槛，锐视测控平台开发了大量基于 .NET 平台的附加功能，包含：易用开源的硬件接口、专业美观的图形界面、功能强大的信号分析、面向应用的设计范例。通过这些附加功能，具有基本 C 语言基础的测控工程师，可以轻松地使用功能强大的 Visual Studio 和 Visual C# 开发环境，创建功能完善的测控系统程序和软件。

获益于 Microsoft® 公司的开源政策，锐视测控平台获得了大量的功能强大的基于 Microsoft® .NET 的原生技术支撑。因此，锐视测控平台不仅将成为业界首个文字语言类专业测控系统开发平台，而且将成为业界唯一一个完全开源开放的专业测控系统开发平台。通过开源开放的锐视测控平台，帮助测控工程师降低测控系统开发入门门槛，提高测控系统开发效率和跨平台移植效率，提高测控系统的可读性、可维护性和可扩展性。

在锐视软件中的 C# 驱动程序是一组 .NET 类库，是在 C# 编程环境中的开发仪器应用程序的编程接口 (API)。C# 驱动程序基于设备的底层驱动程序开发生成，底层驱动程序通常为动态链接库 (DLL)，在 C# 中将 DLL 提供的功能封装为 .NET 类库，生成 C# 驱动，上层应用程序开发人员使用 C# 驱动 (.NET 类库)，底层驱动程序对应用程序开发人员不可见。



锐视驱动的层次结构

(1) 设计要求

对于锐视测控仪器驱动的设计是否合理，从使用者（应用程序开发人员）和开发者（驱动研发团队）两个层面分析。

① 对于驱动使用者（用户，User）：安全可靠高，友好易用，易于应用扩展。

◆ 安全可靠：数据完整性，准确性。三分开发，七分测试。

◆ 友好易用：只封装设备支持的特性，所见即所得，在编程中的自动在线帮助体验

◆ 可扩展性：提供硬件设备的可互换性支持，对于不同设备的相同类型任务，调用方式高度近似。

② 对于驱动开发者（开发者，Developer）：可维护性好，开发效率高，可扩展性强。

就具体实现而言，采用面向对象的设计方法。因为底层驱动程序的灵活性受限，因而总是采用面向过程设计，提供一系列函数，不同的设备差异可能很大，如果不加封装地直接使用开发上层应用，往往会导致开发效率低下、开发难度大、调试以及维护及其不便，程序开发代码如下图所示。

```
//创建模拟信号采集任务
var aitask = new JYPXI62010AITask(0);

//添加采集通道，设置采集模式，采样率和采样点数
aitask.AddChannel(0, -10, 10);
aitask.Mode = AMode.Finite;
aitask.SampleRate = 100000;
aitask.SamplesToAcquire = 5000;

//启动任务并读取数据
var readValue = new double[aitask.SamplesToAcquire * aitask.Channels.Count];
aitask.Start();
aitask.ReadData(ref readValue, -1);
```

示例：在C#中调用筒仪PXI62010驱动程序实现信号采集功能

锐视驱动应用举例

```
//初始化板卡，配置通道参数和触发参数。
uint samplesToAcquire = 1000;
short cardID = NativeDriver.D2K_Register_Card(0, 0);
NativeDriver.D2K_AI_CH_Config((ushort)cardID, 0, NativeDriver.AD_B_10_V);
NativeDriver.D2K_AI_Config((ushort)cardID, NativeDriver.DAQ2K_AI_ADCONVSRV_Int,
NativeDriver.DAQ2K_AI_TRGSRC_ExtD | NativeDriver.DAQ2K_AI_TRGMOD_POST,
samplesToAcquire, 0, 0, true);

//分配存储空间，设定采样率，采集并读取数据。
ushort bufferID;
short[] data = new short[samplesToAcquire];
int sampleRate = 1000000;
uint scanInterval = (uint)(NativeDriver.BaseTime / sampleRate);
NativeDriver.D2K_AI_ContBufferSetup((ushort)cardID, data, (uint)data.Length, out bufferID);
NativeDriver.D2K_AI_ContReadChannel((ushort)cardID, 0, bufferID, samplesToAcquire,
scanInterval, scanInterval, NativeDriver.SYNCH_OP);
```

示例：在C#中调用某模块的直接封装驱动类实现带数字触发的信号采集功能

直接调用底层驱动程序的代码示意图

虚拟仪器 3.0 的驱动设计

采用面向对象的设计，将对象的属性和方法进行封装之后，在锐视驱动的基础上进行应用软件开发的代码如下图所示，由此可见在开发效率提升的同时，可读性、可维护性极大地提高了。

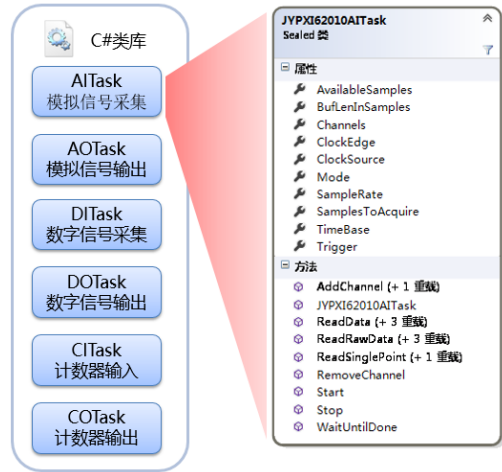
```
//创建模拟信号采集任务
var aitask = new JYPIXI62010AITask(0);

//添加采集通道，设置采集模式，采样率和采样点数和触发方式
aitask.AddChannel(0, -10, 10);
aitask.Mode = AITaskMode.Finite;
aitask.SampleRate = 100000;
aitask.SamplesToAcquire = 5000;
aitask.Trigger.Mode = AITriggerMode.Start;
aitask.Trigger.Type = AITriggerType.Digital;

//启动任务并读取数据
var readValue = new double[aitask.SamplesToAcquire * aitask.Channels.Count];
aitask.Start();
aitask.ReadData(ref readValue, -1);
```

示例：在C#中调用简仪PXI62010驱动类实现带数字触发的信号采集功能

封装为面向对象的锐视驱动之后的代码



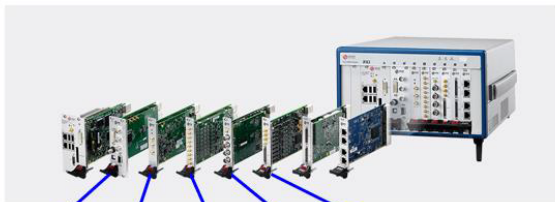
简仪 PXI-62010 多功能数采模块 C# 驱动程序类库

(2) 设计方法

以任务为对象进行功能封装，封装出层次清楚的属性和方法，方便驱动使用者快速搭建测试测量应用。

为每一款设备封装一直专属独立驱动类，易于驱动升级和快速维护更新，不同设备的驱动类相互无影响。优势总结为：

- ① 以任务为对象。快速搭建应用，提供可互换性支持。
- ② 专属设计。简洁易用，可扩展性强。
- ③ 独立自完备。安全可靠，易于维护。



为每一款设备开发专属独立驱动类

以多功能数据采集卡为例，其支持的典型任务分为：模拟采集 (AITask)、模拟输出 (AOTask)、数字采集 (DITask)、数字输出 (DOTask)、计数器输入 (CITask)、计数器输出 (COTask)。

只封装设备支持的特性，借助 Visual Studio 的联想功能从而实现所见即所得式开发过程。

```
aitask.Trigger.Type = AITriggerType.Analog;
aitask.Trigger.Analog.Condition =
    AITriggerCondition.AboveHigh;
//启动任务并读取数据
var readValue = new double[aitask.SamplesToAcquire * aitask.Channels.Count];
aitask.Start();
```

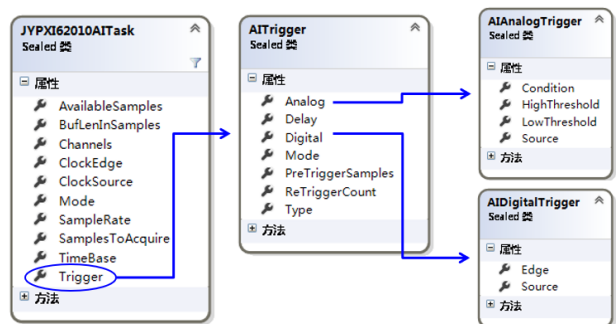
示例：简仪PXI62010模拟信号采集任务的模拟触发支持5种触发条件

```
aitask.Trigger.Type = AITriggerType.Analog;
aitask.Trigger.Analog.Edge = AITriggerEdge.Falling;
//启动任务并读取数据
aitask.Start();
```

示例：简仪PXI62010模拟信号采集任务的模拟触发支持2种触发条件

所见即所得开发过程

以任务为对象，封装层次清楚的属性和方法，以触发为例描述如下：



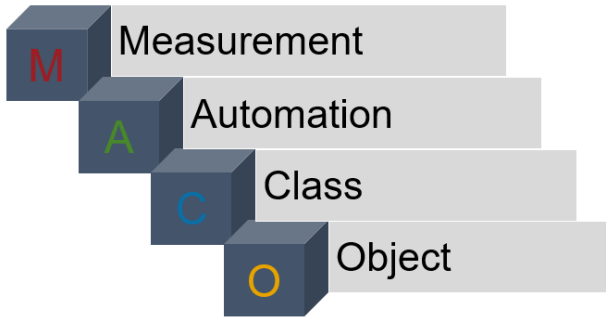
示例：简仪PXI62010模块模拟采集类的触发参数设计

以任务为对象的多层次封装

虚拟仪器 3.0 的驱动设计

(3) MACOs 的概念与创新

MACOs (Measurement Automation Class Objects, 自动化测试测量类和实例库) 是锐视测控平台的软件设计架构, 也是锐视开发软件中的最小编程单位。



MACOs 的寓意

MACOs 在锐视驱动部分的体现为硬件驱动的 MACOs, 以虚拟仪器的基本功能分为 6 个 MACOs, 分别为 (以多功能板卡 JYPXI62205 为例):

- ① 模拟输入 (JYPXI62205AITask)
- ② 模拟输出 (JYPXI62205AOTask)
- ③ 数字输入 (JYPXI62205DITask)
- ④ 数字输出 (JYPXI62205DOTask)
- ⑤ 计数器输入 (JYPXI62205CITask)
- ⑥ 计数器输出 (JYPXI62205COTask)

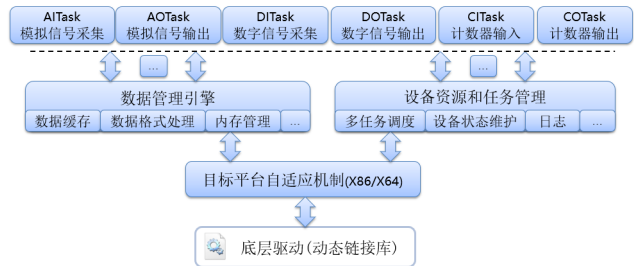
基于 MACOs 架构锐视驱动都是开放型的, 有明确的方法 (Methods) 和属性 (Properties) 定义, 具有如下特点:

- ① 不同厂家可以轻松地实现其它厂家的功能;
- ② 所有的功能都是由 OOP 的方法和属性决定的;
- ③ 用户无需关心底层的硬件驱动, 只需关心上层的方法和属性即可;

- ④ 大大地降低了硬件互通性的难度, 也摆脱了少数大公司控制标准的束缚;
- ⑤ 直接用微软的编译, 无需经过图形编译;
- ⑥ 极大地提高了效率和可维护性。

(4) 锐视驱动的层次构成

采用 MACOs 架构的锐视驱动层次构成如下图所示:



锐视驱动层次构成

示例和拓展

基于锐视驱动开发数据采集软件, 典型的操作都是如下图所示的 5 步。



基于锐视驱动的有限点数据采集流程

```
//创建模拟信号采集任务  
var aitask = new CTPXI62010AITask(0);  
//添加采集通道, 设置采集模式, 采样率和  
aitask.AddChannel(0, -10, 10);  
aitask.Mode = AllMode.Finite;  
aitask.SampleRate = 100000;  
aitask.SamplesToAcquire = 5000;  
//启动任务并读取数据  
var readValue = new double[aitask.Samp  
aitask.Start();  
aitask.ReadData(ref readValue, -1);  
  
//创建模拟信号采集任务  
var aitask = new CTPXI62005AITask(0);  
//添加采集通道, 设置采集模式, 采样率和采集点数  
aitask.AddChannel(0, -10, 10);  
aitask.Mode = AllMode.Finite;  
aitask.SampleRate = 100000;  
aitask.SamplesToAcquire = 5000;  
//启动任务并读取数据  
var readValue = new double[aitask.SamplesToAcquire * aitask.Channels.Count];  
aitask.Start();  
aitask.ReadData(ref readValue, -1);
```

示例: 在C#中调用筒仪PXI62010驱动实现有限点信号采集功能

示例: 在C#中调用筒仪PXI62005驱动实现有限点信号采集功能

同类板卡相同功能快速替换

虚拟仪器 3.0 的界面设计

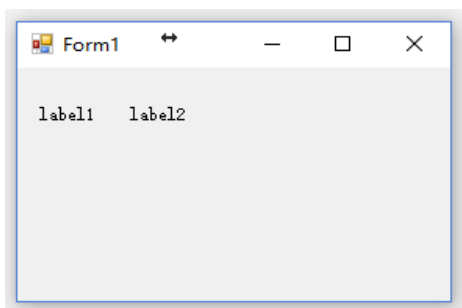
在界面设计中最重要就是控件以及布局，如果将其比喻成为厨师做饭的过程，那么控件就是厨师做饭所用的原材料，而布局就是厨师做饭所用的烹饪手法。

控件

(1) C# 中常见的控件及功能

① Label 控件

在界面设计中主要用于标注提示的控件



常用属性：

◆ Text 属性：用来设置或返回标签控件中显示的文本信息。

◆ BackColor 属性：

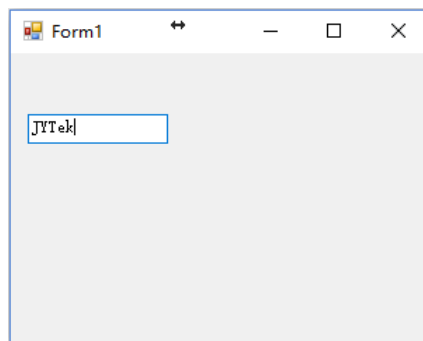
用来获取或设置控件的背景色。当该属性值设置为 Color.Transparent 时，标签将透明显示，即背景色不再显示出来。

◆ BorderStyle 属性：用来设置或返回边框。有三种选择：BorderStyle.None 为无边框（默认），BorderStyle.FixedSingle 为固定单边框，BorderStyle.Fixed3D 为三维边框。

◆ Enabled 属性：用来设置或返回控件的状态。值为 true 时允许使用控件，值为 false 时禁止使用控件，此时标签呈暗淡色，一般在代码中设置。标签还具有 Visible、ForeColor、Font 等属性，具体含义请参考窗体的相应属性。

② TextBox 控件

在界面设计中主要用于显示或者接受用户输入的文本信息。



主要属性：

◆ Text 属性：Text 属性是文本框最重要的属性，因为要显示的文本就包含在 Text 属性中。默认情况下，

最多可在一个文本框中输入 2048 个字符。如果将 MultiLine 属性设置为 true，则最多可输入 32KB 的文本。

Text 属性可以在设计时使用【属性】窗口设置，也可以在运行时用代码设置或者通过用户输入来设置。可

以在运行时通过读取 Text 属性来获得文本框的当前内容。

◆ MaxLength 属性：用来设置文本框允许输入字符的最大长度，该属性值为 0 时，不限制输入的字符

数。

◆ MultiLine 属性：用来设置文本框中的文本是否可以输入多行并以多行显示。值为 true 时，允许多行显示。值为 false 时不允许多行显示，一旦文本超过文本框宽度时，超过部分不显示。

◆ HideSelection 属性：用来决定当焦点离开文本框后，选中的文本是否还以选中的方式显示，值为 true，则不以选中的方式显示，值为 false 将依旧以选中的方式显示。

◆ ReadOnly 属性：用来获取或设置一个值，该值指示文本框中的文本是否为只读。值为 true 时为只读，

值为 false 时可读可写。

◆ PasswordChar 属性：是一个字符串类型，允许设置一个字符，运行程序时，将输入到 Text 的内容全部显示为该属性值，从而起到保密作用，通常用来输入口令或密码。

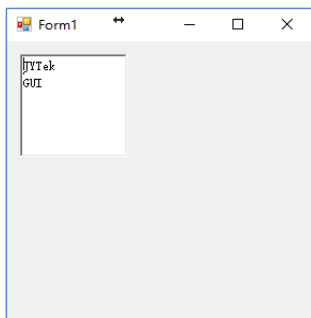
虚拟仪器 3.0 的界面设计

◆ ScrollBars 属性：用来设置滚动条模式，有四种选择：ScrollBars.None(无滚动条)，ScrollBars.Horizontal(水平滚动条)，ScrollBars.Vertical(垂直滚动条)，ScrollBars.Both(水平和垂直滚动条)。

注意：只有当 MultiLine 属性为 true 时，该属性值才有效。在 WordWrap 属性值为 true 时，水平滚动条将不起作用。

③ RichTextBox 控件

RichTextBox 是一种既可以输入文本、又可以编辑文本的文字处理控件，与 TextBox 控件相比，RichTextBox 控件的文字处理功能更加丰富，不仅可以设定文字的颜色、字体，还具有字符串检索功能。另外，RichTextBox 控件还可以打开、编辑和存储 .rtf 格式文件、ASCII 文本格式文件及 Unicode 编码格式的文件。



常用属性：

上面介绍的 TextBox 控件所具有的属性，RichTextBox 控件基本上都具有，除此之外，该控件还具有一些其他属性。

◆ RightMargin 属性：用来设置或获取右侧空白的大小，单位是像素。通过该属性可以设置右侧空白，

如希望右侧空白为 50 像素，可使用如下语句：

```
RichTextBox1.RightMargin=RichTextBox1.Width-50;
```

插入点处。如果选定了文本，则给该属性所赋的文本将替换掉选定文本。

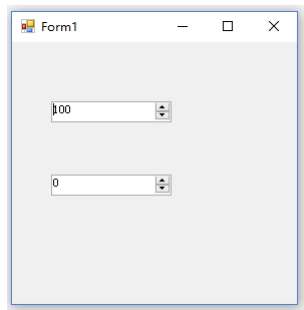
◆ SelectionColor 属性：用来获取或设置当前选定文本或插入点处的文本颜色。

◆ SelectionFont 属性：用来获取或设置当前选定文本或插入点处的字体。

④ NumericUpDown 控件

控件组中的 NumericUpDown 控件看起来像是一个文本框与一对用户可单击以调整值的箭头的组合。可以通过单击向上和向下按钮、按向上和向下箭头键来增大和减小数字，也可以直接输入数字。单击向上箭头键时，值向最大值方向增加；单击向下箭头键时，值向最小值方向减少。该控件在工具箱中的图标为。

该控件的样式如图下图所示。



常用属性：

◆ DecimalPlaces：获取或设置该控件中显示的小数位数。

◆ Hexadecimal：获取或设置一个值，该值指示该控件是否以十六进制格式显示所包含的值。

◆ Increment：获取或设置单击向上或向下按钮时，该控件递增或递减的值。

◆ Maximum：获取或设置该控件的最大值。

◆ Minimum：获取或设置该控件的最小值。

◆ Value：获取或设置该控件的当前值。

与 TextBox 控件一样，NumericUpDown 控件的常用事件有：ValueChanged、GotFocus、LostFocus 等。

⑤ Button 控件

Button 控件又称按钮控件，是 Windows 应用程序中最常用的控件之一，通常用它来执行命令。如果按钮具有焦点，就可以使用鼠标左键、Enter 键或空格键触发该按钮的 Click 事件。通过设置窗体的 AcceptButton 或 CancelButton 属性，无论该按钮是否有焦点，都可以使用户通过按 Enter 或 Esc 键来触发按钮的 Click

虚拟仪器 3.0 的界面设计

事件。一般不使用 Button 控件的方法。Button 控件也具有许多如 Text、ForeColor 等的常规属性，此处不再介绍，只介绍该控件有特色的属性。以后介绍的控件也采用同样的方法来处理。

常用属性：

◆ FlatStyle 属性：用来设置按钮的外观。

常用事件：


◆ Click 事件：当用户用鼠标左键单击按钮控件时，将发生该事件。

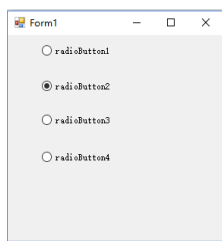
◆ MouseDown 事件：当用户在按钮控件上按下鼠标按钮时，将发生该事件。

◆ MouseUp 事件：当用户在按钮控件上释放鼠标按钮时，将发生该事件。

⑥ RadioButton 控件

RadioButton 又称单选按钮，其在工具箱中的图标为

 **RadioButton**，单选按钮通常成组出现，用于提供两个或多个互斥选项，即在一组单选按钮中只能选择一个。



常用属性：

◆ Checked 属性：用来设置或返回单选按钮是否被选中，选中时值为 true，没有选中时值为 false。

◆ AutoCheck 属性：如果 AutoCheck 属性被设置为 true（默认），那么当选择该单选按钮时，将自动清除该组中所有其他单选按钮。对一般用户来说，不需改变该属性，采用默认值（true）即可。

◆ Appearance 属性：用来获取或设置单选按钮控件的

外观。当其取值为 Appearance.Button 时，将使单选按钮的外观像命令按钮一样：当选定它时，它看似已被按下。当取值为 Appearance.Normal 时，就是默认的单选按钮的外观。

◆ Text 属性：用来设置或返回单选按钮控件内显示的文本，该属性也可以包含访问键，即前面带有“&”符号的字母，这样用户就可以通过同时按 Alt 键和访问键来选中控件。

常用事件：

◆ Click 事件：当单击单选按钮时，将把单选按钮的 Checked 属性值设置为 true，同时发生 Click 事件。

◆ CheckStateChanged 事件：当 Checked 属性值更改时，将触发 CheckStateChanged 事件。

⑦ CheckBox 控件

CheckBox 控件的常用属性如下。

◆ ThreeState 属性：用来返回或设置复选框是否能表示三种状态，如果属性值为 true 时，表示可以表示三种状态——选中、没选中 and 中间态（CheckState.Checked、CheckState.Unchecked 和 CheckState.Indeterminate），属性值为 false 时，只能表示两种状态——选中 and 没选中。

◆ Checked 属性：用来设置或返回复选框是否被选中，值为 true 时，表示复选框被选中，值为 false 时，

表示复选框没被选中。当 ThreeState 属性值为 true 时，中间态也表示选中。

⑧ Timer 控件

Timer 控件又称定时器控件或计时器控件，在工具箱中的图标是，该控件的主要作用是按一定的时间间隔周期性地触发一个名为 Tick 的事件，因此在该事件的代码中可以放置一些需要每隔一段时间重复执行的程序段。在程序运行时，定时器控件是不可见的。

常用属性：

◆ Enabled 属性：用来设置定时器是否正在运行。值为 true 时，定时器正在运行，值为 false 时，定时器不在运行。

虚拟仪器 3.0 的界面设计

◆ Interval 属性：用来设置定时器两次 Tick 事件发生的时间间隔，以毫秒为单位。如它的值设置为 500，则将每隔 0.5 秒发生一个 Tick 事件。

常用方法：

◆ Start 方法：用来启动定时器。调用的一般格式如下：

Timer 控件名 .start(); 该方法无参数。

◆ Stop 方法：用来停止定时器。调用的一般格式如下：

Timer 控件名 .stop(); 该方法无参数。

常用事件：

定义器控件响应的事件只有 Tick，每隔 Interval 时间后将触发一次该事件。

目前锐视测控平台中的 JY.GUI 类库目前包含 10 多种测试测量领域中常用 GUI，这边给大家做一个简单的介绍。

(1) EasyChart 类

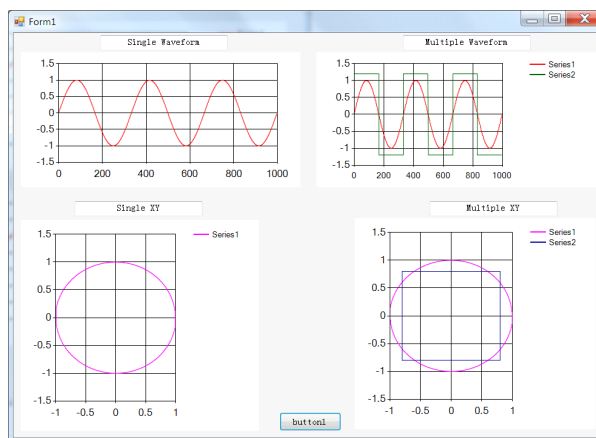
EasyChart 为基于微软 MSChart 控件实现的自定义控件，封装了最常用的单通道或多通道连续波形、单通道或多通道离散波形显示功能，包含的属性和方法如下表所示：

属性	
属性名称	功能描述
LegendVisible	设定是否在 Chart 中显示图例，默认为 True。
Palette	设定一系列颜色，依次用于 Chart 中各通道波形的颜色显示。
XAxisLogarithmic ¹	设定 X 轴是否采用对数显示，默认为 False。
YAxisLogarithmic ¹	设定 Y 轴是否采用对数显示，默认为 False。
方法	
Plot	显示单通道或多通道连续波形、单通道或多通道离散波形。

注 1：该属性在运行时不可设置。该属性设为 True 之后，调用 Plot 方法时输入的相应数据

在 Windows 窗体上放置 EasyChart 控件后，可以选中该 EasyChart 控件后在属性表格中对上述属性进行设置并查看其效果，除非特别说明，这些属性也可以在运行时设置。

EasyChart 支持单通道或多通道的连续或离散波形显示。此处的连续波形是指波形上任意相邻的两个点在 X 轴上均有相同的间隔，通常用于时域采样波形或频谱显示，在调用 Plot 方法显示连续波形时，除了输入波形数据，还可设定 X 轴上的起始位置和间隔；而离散波形是指波形上的任意一个点，都由独立的一对 x 和 y 来确定其在 Chart 上的位置，在调用 Plot 方法显示离散波形时，必须输出等长的两个数组，一一对应各点在 Chart 上的位置。详情可参看 EasyChart 的例程，下图是 EasyChart 例程的运行效果图，4 个图分别对应单通道连续波形、多通道连续波形、单通道离散波形和多通道离散波形。



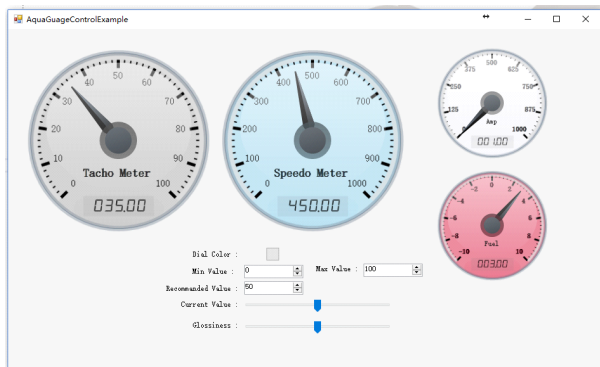
(2) AquaGauge 类

AquaGauge 是一个常用的仪表盘：

属性	
属性名称	功能描述
BackColor	配置表盘背景颜色。
Glossiness	反光镜效果调试比例 (0<=&&<=100 的数值)
Max	表盘最大值
Min	表盘最小值
TextDescription	表盘描述说明
NumberOfDivisions	表盘分隔个数，数值范围 (1<=&&<=25 的数值)
NumberOfSubDivisions	表盘子分隔个数，数值范围 (1<=&&<=10 的数值)
Value	表盘所显示的当前值

整体使用效果如下图所示，详见请参考配套范例，AquaGaugeExample，对控件使用有更加直观的参考。

虚拟仪器 3.0 的界面设计

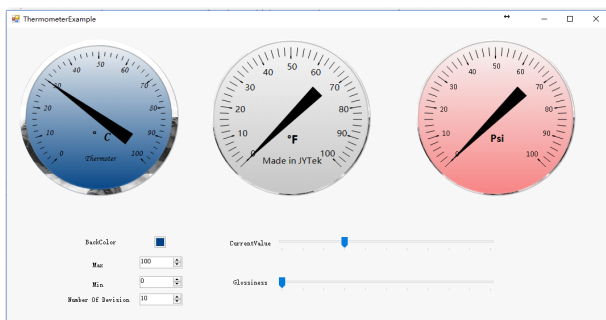


(3) Thermoter 类

Thermoter 同样是一个仪表盘控件，具有另一种风格的控件：

属性	
属性名称	功能描述
BackColor	配置表盘背景颜色
BorderWidth	配置表盘边框厚度
Glossiness	反光镜效果调试比例 (0<=&&=100 的数值)
Max	表盘最大值
Min	表盘最小值
DescriptionText	表盘描述说明
UnitText	表盘单位描述
NumberOfDivisions	表盘分隔个数，数值范围 (1<=&&=25 的数值)
NumberOfSubDivisions	表盘子分隔个数，数值范围 (1<=&&=10 的数值)
Value	表盘所显示的当前值

整体使用效果如下图所示，详见请参考配套范例，ThermoterExample，对控件使用有更加直观的参考。

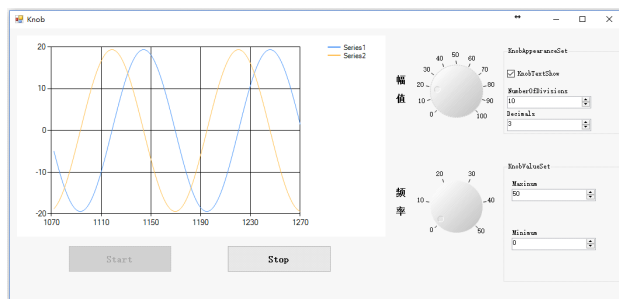


(4) Knob 类

Knob 类是一个常用的旋钮控件，用该控件完成数值的输入操作，控件常用的属性方法如下表所示。

属性	
属性名称	功能描述
NumberOfDigits	数值显示时科学表达法的有效位数 (0<=&&=10 的数值)
Max	旋钮最大值
Min	旋钮最小值
NumberOfDivisions	旋钮分隔个数，数值范围 (1~25 的数值)
TickVisible	Bool 参数，是否显示旋钮数值
Value	旋钮所表示的当前值
事件	
ValueChanged	默认事件

整体使用效果如下图所示，详见请参考配套范例，KnobExample，对控件使用有更加直观的参考。



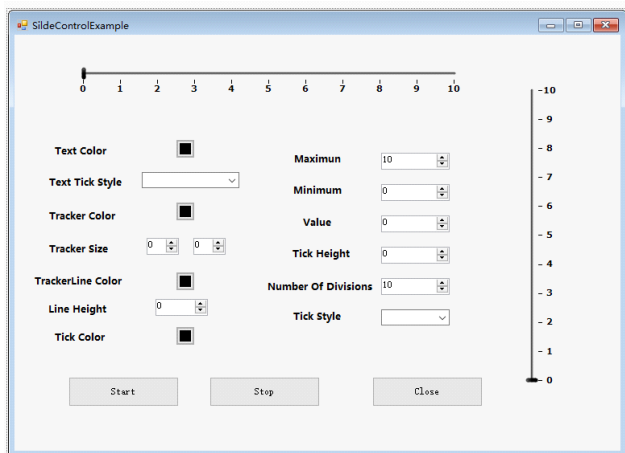
(5) Slider 类

Slider 类所表示的是一个常用的滑杆控件，用该控件完成数值的输入操作，控件常用的属性方法如下表所示。

属性	
属性名称	功能描述
NumberOfDigits	数值显示时科学表达法的有效位数 (0<=&&=10 的数值)
Max	滑动杆最大值
Min	滑动杆最小值
NumberOfDivisions	滑动杆分隔个数，数值范围 (1<=&&=25 的数值)
Orientation	滑动杆方向 (Vertical, Horizontal) 两种方向的选择
TextStyle	滑动杆数值表示风格 (None, TopLeft, BottomRight, Both)
ForeColor	滑动杆 Text 颜色选择
TickStyle	滑动杆 Tick 表示风格 (None, TopLeft, BottomRight, Both)
TrickHeight	滑动杆 Tick 高度选择
TrickColor	滑动杆 Trick 颜色选择
TrackerColor	滑动块颜色选择
TrackerSize	滑动杆滑块大小，如果 with 与 Height 相等，则会成为圆形
LineColor	滑动杆主干线颜色选择
LineWidth	滑动杆主干线高度选择
Value	滑动杆所表示的当前值
事件	
ValueChanged	默认事件

虚拟仪器 3.0 的界面设计

整体使用效果如下图所示，详见请参考配套范例，SlideExample，对控件使用有更加直观的参考。



(6) Led 类

Led 类所表示的是一个常用布尔灯控件，用该控件完成 bool 数值的显示操作，控件常用的属性方法如下表所示。

属性	
属性名称	功能描述
BlinkColor	灯闪烁的颜色
BlinkInterval	灯光闪烁时长，单位毫秒（100<=且<=5000 的数值）
BlinkOn	Bool 参数，是否打开闪烁功能
OffColor	灯光关闭时的颜色选择
OnColor	灯光打开时的颜色选择
Style	灯样式选择，Circular, Rectangular
Value	Bool 参数，灯光开闭选择

(7) LedBright 类

LedBright 类所表示的是另一种的布尔灯控件，用该控件完成 bool 数值的显示操作，控件常用的属性方法如下表所示。

属性	
属性名称	功能描述
BlinkColor	灯闪烁的颜色
BlinkInterval	灯光闪烁时长，单位毫秒（100<=且<=5000 的数值）
BlinkOn	Bool 参数，是否打开闪烁功能
Color	灯光打开时的颜色选择
Style	灯样式选择，Circular, Rectangular
Value	Bool 参数，灯光开闭选择

(8) EasyButton 类

EasyButton 类所表示的是 Button 类一个衍生类，可以预设一些图片在 Button 控件上面。

属性	
属性名称	功能描述
PreSetImage	进行 Button 按钮图片预设功能

(9) IndustrySwitch 类

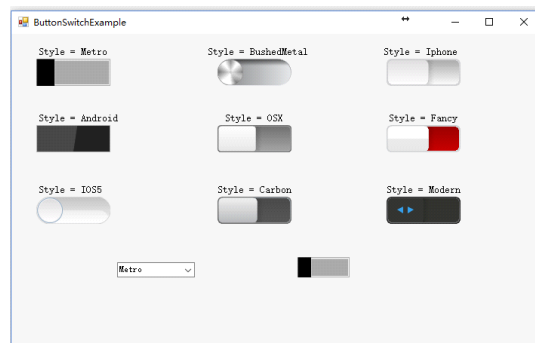
IndustrySwitch 类表示一个工业开关的一个类，具体属性如下表所示，上述几个控件类有一个操作范例 SwitchControlExample，更多的内容请参考范例内容进行学习。

属性	
属性名称	功能描述
Style	多种工业 Switch 复选功能
Value	Bool 参数，Switch 开关选择
事件	
ValueChanged	默认事件，值改变事件

(10) ButtonSwitch 类

ButtonSwitch 类表示一个按钮开关的一个类，具体属性如下表所示，详见请参考配套范例，ButtonSwitchExample，对控件使用有更加直观的参考。

属性	
属性名称	功能描述
Style	多种 Switch 复选功能
Value	Bool 参数，Switch 开关选择
事件	
ValueChanged	默认事件，值改变事件

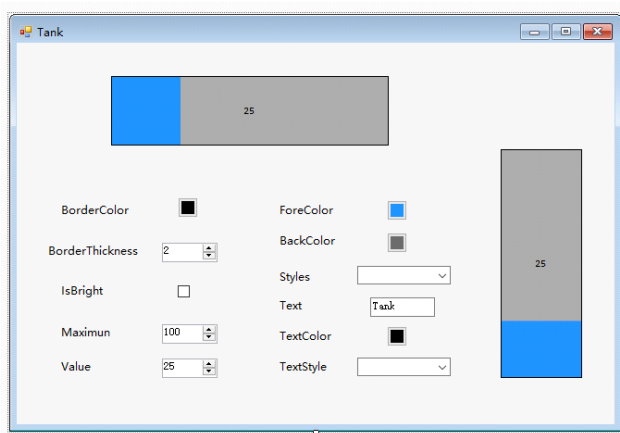


虚拟仪器 3.0 的界面设计

(11) Tank 类

Tank 类表示数值属性，表示一种水罐的控件类，具体属性如下表所示，详见请参考配套范例，TankExample，对控件使用有更加直观的参考。

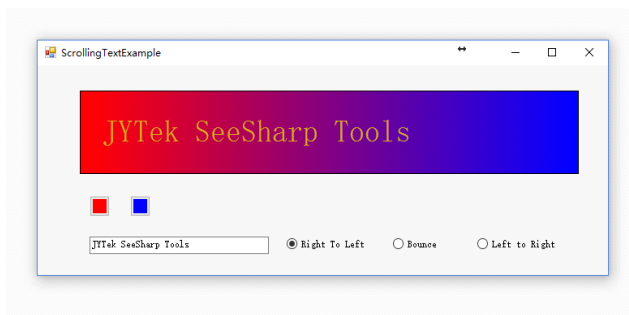
属性	
属性名称	功能描述
BorderColor	边框颜色选择
BorderWidth	边框厚度选择
IsBright	Bool 参数，是否选用亮色显示 Tank 控件
Maximun	Tank 数值最大值
Orientation	Tank 摆放方向
Styles	Tank 样式选择，Solid,Dashed
TextColor	Tank 文本显示颜色选择
TextStyle	Tank 文本显示样式选择 (None, Percentage, Text, Value, ValueOverMaximum)
Value	Tank 所表示的当前值



(12) ScrollingText 类

ScrollingText 类表示一个字符串的展示，具体属性如下表所示，详见请参考配套范例，ScrollingText，对控件使用有更加直观的参考。

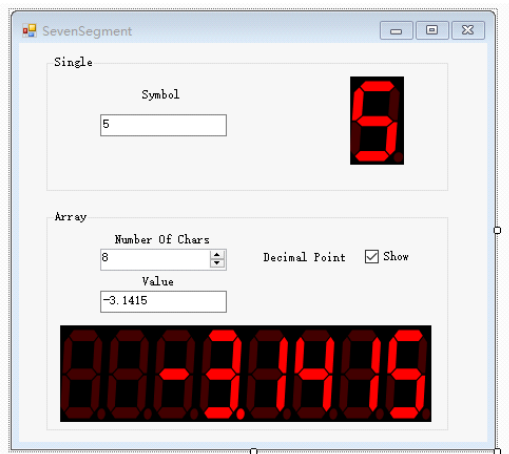
属性	
属性名称	功能描述
BorderColor	边框颜色选择
BorderThickness	边框厚度选择
ScrollDirection	文本滚动方向选择
BorderVisible	Bool 参数，是否需要显示边框
Text	滚动说明文本
SrollSpeed	滚动文本速度 (1<=&&&=1000 的数值)
VerticleAligment	设置滚动条文字位置



(13) SevenSegment 类

SevenSegment 类表示一个七段数码管，具体属性如下表所示，详见请参考配套范例，SevenSegment，对控件使用有更加直观的参考。

属性	
属性名称	功能描述
BackColor	背景颜色
DarkColor	数码管暗时颜色
LightColor	数码管亮时颜色
DecimalShow	是否显示小数点位置
ItalicFactor	数码管倾斜系数
NumberOfChar	数码管总个数
Value	数码管显示内容

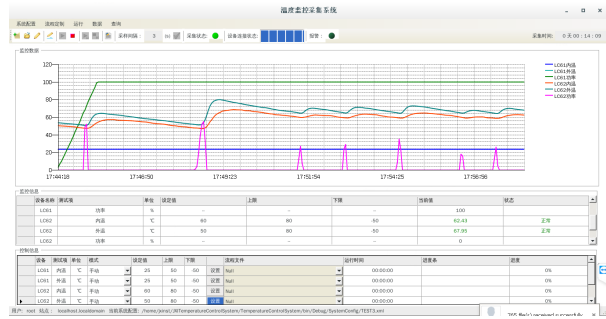


(14) SevenSegmentBright 类

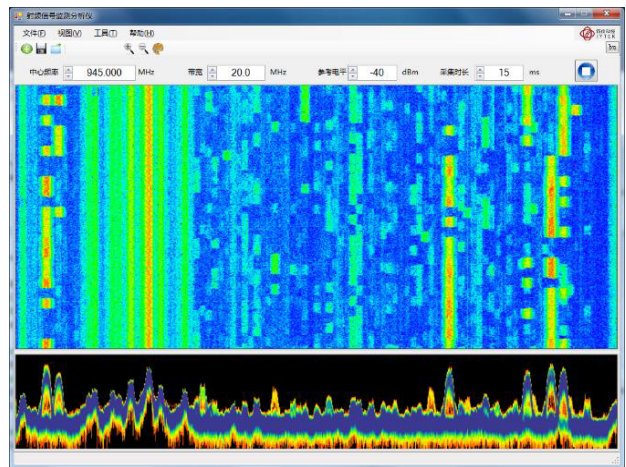
SevenSegmentBright 类表示一个七段数码管，对比而言有更多的颜色选择，具体属性如下表所示，详见请参考配套范例，SevenSegmentBright，对控件使用有更加直观的参考。

虚拟仪器 3.0 的界面设计

属性名称	功能描述
BackColor1	背景颜色 1
BackColor2	背景颜色 2，由背景颜色 1 与背景颜色 2 共同混合为新的背景色
GradientBackground	是否显示渐变背景色效果
BorderColor	边框颜色
BorderWith	边框宽度
DarkColor	数码管不显示时颜色
ForColor	数码管显示时的颜色选择
HighlightOpaque	显示半透明颜色，默认值为 50
NumberOfChar	数码管总个数



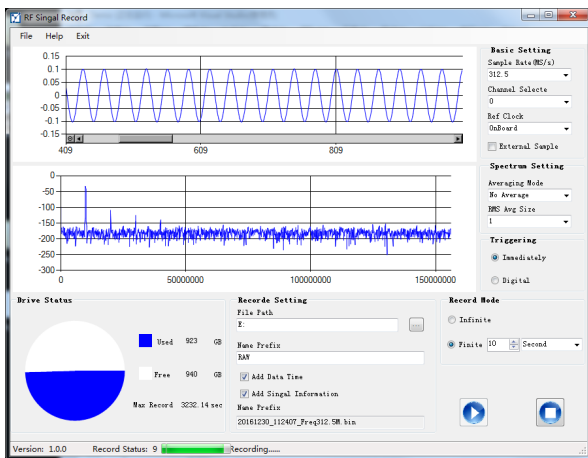
温度控制界面以及显示



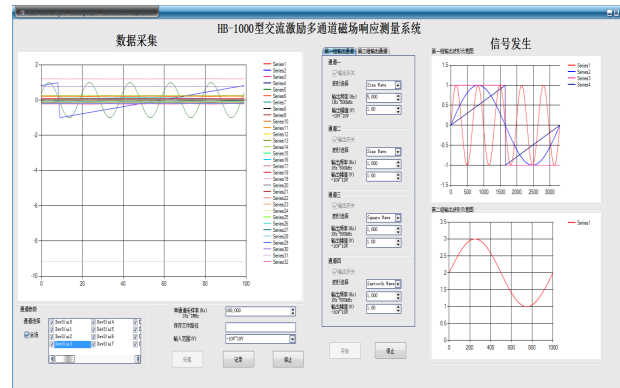
无线电监测系统

界面设计参考

当我们了解了整个界面控件组成，更多的就是对于界面的设计，针对于界面设计本身那就仁者见仁，智者见智了。不过我觉得可以提供一些在 SeeSharp 平台下面做的一些优秀的界面作为例子，仅供参考。



导航信号高速流盘系统



交流激励多通道磁场响应测量系统

虚拟仪器 3.0 的分析工具

随着计算机技术高效、快速的数据处理功能越来越强大，基于微型计算机的虚拟仪器技术以其传统仪器所无法比拟的强大的数据采集、分析、处理、显示和存储功能得到了广泛应用，显示出其强劲的生命力。与传统的仪器不同，虚拟仪器可使用相同的硬件系统，通过不同的软件就可以实现功能完全不同的各种测量测试仪器。虚拟仪器技术最核心的思想，就是利用计算机的硬/软件资源，使本来需要硬件实现的技术软件化（虚拟化），以便最大限度地降低系统成本，增强系统的功能与灵活性。在虚拟仪器技术中，分析工具无疑是极为重要的。工程师们可以利用强大的 PC 技术，结合各种开发环境对海量数据进行实时或者离线的分析和处理。下面将主要介绍在虚拟仪器 3.0 技术中我们可以使用的一些分析工具，包括数学分析、信号分析、图像分析、音频分析等。

信号分析

(1) 锐视工具 (SeeSharp Tools)

锐视测控平台是由简仪科技自主开发的强大、易用、开源的测控系统开发平台。借助于 Microsoft® .NET 平台和 Microsoft® Visual Studio 开发环境的强大技术支撑，锐视测控平台将成为业界第一个功能强大、且完全开源的测控系统专业开发平台。锐视测控平台提供基于 Visual C# 语言和 x86 架构的一系列软硬件解决方案，使得仅具有基本 C 语言基础的测控工程师也可以轻松快速地开发出功能强大、界面专业、易于维护和扩展的测控系统，从而大大提高测控系统的开发效率，实现软硬件之间的无缝连接。

锐视工具中包含了大量的算法工具，并且仍然在不断丰富和发展中。锐视工具中有些算法基于 .NET 基类实现，有些高级算法基于 Intel 的 MKL 类库。Intel 数学核心函数库 (MKL) 是一套高度优化、线程安全的数学例程、函数，面向高性能的工程、科学与财务应用。锐视工具目前包含的主要算法如下：

- ① 常用数组运算和操作：数组转置、索引替换、RMS 计算等；
- ② 常用波形和频谱生成：标准、白噪声波形生成；
- ③ 数字滤波器：FIR 滤波器，IIR 滤波器，各种窗函数；
- ④ 信号分析：频谱分析，谐波分析，时域分析，相位差计算等。

(2) MATLAB

目前，MATLAB 已经发展成为适合多学科的大型软件。在世界各高校，MATLAB 已经成为线性代数、数值分析、数理统计、优化方法、自动控制、数字信号处理、动态系统仿真等高级课程的基本教学工具。但对于虚拟仪器开发来说，MATLAB 还是有不足之处的。例如界面开发能力较差，并且数据输入、网络通信、硬件控制等方面都比较繁琐。因此，很少有用 MATLAB 单独开发虚拟仪器的。通常都将其作为一种辅助工具，和其它语言结合起来，通过混合编程进行虚拟仪器的开发。

MATLAB 和 C# 的混合编程，只需要简单的三个步骤，基本思路就是先在 MATLAB 编写包含算法的 .m 文件，然后将 MATLAB 函数打包成 DLL 文件，最后在 C# 中调用这个 DLL 就可以了。整个过程很简单，可移植性较好，并且目标机器不需要运行 MATLAB 环境，只需要安装一个免费的 MATLAB 运行时引擎 (MCR) 即可。

MATLAB 与 C# 的混合编程，结合了 C# 强大的界面开发、网络通信、硬件控制的优势以及 MATLAB 强大的信号处理以及数学运算的优势，使其成为虚拟仪器 3.0 技术实现的重要方法之一。

(3) Measurement Studio

Measurement Studio 是美国国家仪器 (NI) 是为 Visual Studio .NET 和 Visual Studio 6.0 环境提供的一个集成式套件，包括各种常用的测量和自动化控件、工具和类库。NI Measurement Studio 带有的 ActiveX 和 .NET 控件、面向对象的测量硬件接口、高级的分析库、科学的用户界面控件、测量数据网络化、向导、交互式代码设计器和高扩展性类库等功能，极大的减少了您应用程序的开发时间。

数学分析

(1) Math.NET

工程技术人员不可避免的会涉及到数值计算程序的编制，当前也存在一些功能非常强大的数值计算库，如 IMSL 等，然而这些商用库通常都不便宜。Math.NET 是开源的数值计算库，完全免费。C# 用于 Windows 桌面程序的开发，有着无可比拟的优势，无论是速度

虚拟仪器 3.0 的分析工具

上、语言易学性上。而 Math.NET 是完全基于 C# 编写，不会存在兼容性的问题。因此，对于只是开发小型的非商业的数值计算程序的人来说，利用 C# 与 Math.NET 的联合，无疑是首要选择。

Math.NET 的初衷是开源建立一个稳定并持续维护的先进的基础数学工具箱，以满足 .NET 开发者的日常需求。目前该组件主要分为以下几个子项目，该组件同时也支持 Mono，而且支持的平台也非常广泛 (PCL Portable Profile 47: Windows 8, Silverlight 5, Xamarin: Android, iOS)。

① Math.NET Numerics 是核心功能是数值计算。主要是提供日常科学工程计算相关的算法，包括一些特殊函数，线性代数，概率论，随机函数，微积分，插值，最优化等相关计算功能。它支持的主要特征有：

- ◆ 支持概率分布：离散型、连续型和多元；
- ◆ 伪随机数生成器；
- ◆ 支持稀疏矩阵和向量的复杂的线性代数解决方法；
- ◆ LU, QR, SVD, EVD, Cholesky 分解；
- ◆ 矩阵读写功能，支持 Matlab 和一些分开的文件；
- ◆ 复数计算；
- ◆ 特殊函数：Gamma, Beta, Erf, Bessel, Struve 等等；
- ◆ 插值，线性回归，曲线拟合；
- ◆ 数值积分，方程求解；
- ◆ 描述性统计、统计直方图，皮尔森相关系数；
- ◆ 马尔可夫链蒙特卡罗抽样；
- ◆ Mono 平台支持，可选支持英特尔数学内核库 (Microsoft Windows 和 Linux)；

② Math.NET Filtering 是一个数字信号处理工具箱，提供了数字滤波器的基础功能，以及滤波器应用到数字信号处理和数据流转换的相关功能。

(2) Meta.Numerics

Meta.Numerics 是一个 Mono 兼容的 .NET 开发包，用于科学数学计算编程。它包括矩阵代数（包括

SVD，非对称 eigensystems 的，和稀疏矩阵），特殊功能的实数和复数（包括贝塞尔函数和复杂的误差函数）的功能，统计和数据分析（包括 PCA，物流和非线性回归，统计测试，和不均匀随机偏离），和信号处理（包括任意长度的 FFT）。Meta.Numerics 是完全基于面向对象的类库并且在执行速度上做了大量的优化，可以在 C#、Visual Basic、F# 以及其它的 .NET 环境中使用。

图像处理和音视频分析

(1) Accord.NET (AForge.NET)

AForge.NET 是一个专门为开发者和研究者基于 C# 框架设计的，包括计算机视觉与人工智能，图像处理，神经网络，遗传算法，机器学习，模糊系统，机器人控制等领域。该框架架构合理，易于扩展，涉及多个较前沿的技术模块，可以为相关开发人员或科研人员的工作提供极大便利。Accord.NET 其实是 AForge.NET 框架的扩展，在其基础上提供了更多统计分析和处理函数，包括图像处理和计算机视觉算法，它遵从 LGPL 授权协议。Accord.NET 框架主要有三个大的功能性模块，分别为科学技术、信号与图像处理和支撑组件。

① 科学计算

- ◆ Accord.Math: 包括矩阵扩展程序，以及一组矩阵数值计算和分解的方法，也包括一些约束和非约束问题的数值优化算法，还有一些特殊函数以及其他一些辅助工具。
- ◆ Accord.Statistics: 包含概率分布、假设检验、线性和逻辑回归等统计模型和方法，隐马尔科夫模型，(隐藏) 条件随机域、主成分分析、偏最小二乘判别分析、内核方法和许多其他相关的技术。
- ◆ Accord.MachineLearning: 为机器学习应用程序提供包括支持向量机，决策树，朴素贝叶斯模型，k-means 聚类算法，高斯混合模型和通用算法如 Ransac, 交叉验证和网格搜索等算法。

- ◆ Accord.Neuro: 包括大量的神经网络学习算法，如 Levenberg-Marquardt, Parallel Resilient Backpropagation, Nguyen-Widrow 初始化算法，深层的信念网络和许多其他神经网络相关的算法。

虚拟仪器 3.0 的分析工具

② 信号与图像处理

◆ Accord.Imaging: 包含特征点探测器 (如 Harris, SURF, FAST and FREAK), 图像过滤器、图像匹配和图像拼接方法, 还有一些特征提取器。

◆ Accord.Audio: 包含一些机器学习和统计应用程序需要的处理、转换过滤器以及处理音频信号的方法。

◆ Accord.Vision: 实时人脸检测和跟踪, 以及对人流图像中的一般的检测、跟踪和转换方法, 还有动态模板匹配追踪器。

③ 支持组件, 主要是为上述一些组件提供数据显示, 绘图的控件, 分为以下几个命名空间:

◆ Accord.Controls: 包括科学计算应用程序常见的柱状图、散点图和表格数据浏览。

◆ Accord.Controls.Imaging: 包括用来显示和处理的图像的 WinForm 控件, 包含一个方便快捷显示图像的对话框。

◆ Accord.Controls.Audio: 显示波形和音频相关性信息的 WinForm 控件。

◆ Accord.Controls.Vision: 包括跟踪头部, 脸部和手部运动以及其他计算机视觉相关的任务 WinForm 控件。

虚拟仪器 3.0 的应用领域

引言：基于虚拟仪器 3.0 的测控系统采用标准化硬件平台技术，借助锐视测控平台的强大，易用，开源特性，可以完成复杂的测试和控制项目，接下来会介绍在各个领域的项目案例。

总控系统

(1) 概述

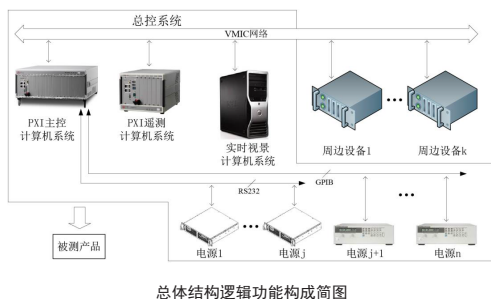
总控系统作为仿真系统的控制中心，协调、调度整个半实物仿真系统运行，除了通过计算机系统完成仿真系统的任务调度和时序控制以外，还要完成对产品的供电、时序逻辑控制及产品电气接口功能，如 GJB289A 总线通讯、模拟发送飞行任务和一次性指令、ARINC-429 总线通讯、遥测数据采集等；此外，总控系统还包含仿真实时网络系统，用于仿真系统内部各个设备间仿真数据和指令信息的实时交互。

(2) 系统组成与工作原理

总控系统由三个计算机测控子系统组成，其分别为：主控计算机系统、遥测计算机系统和实时视景计算机系统。主控计算机系统和遥测计算机系统采用 PXI 总线高性能工控机，而实时视景计算机系统采用高性能图形工作站。

三个计算机系统通过星型 VMIC 网络及 RS232 和 GPIB 总线，与周边设备（如实时仿真单元、高速数据链单元、目标模拟器单元以及电源单元等）组成一个实时、网络化计算机仿真测试平台。

总控系统通过调理单元与被测型号产品进行电气信号互联，并对周边设备进行控制，最终实现控制、仿真、测试等核心功能。总控系统的总体结构逻辑功能构成如下图所示。



总体结构逻辑功能构成简图

其中，主控计算机系统和遥测计算机系统为高性能 PXI 总线计算机系统；而实时视景计算机系统选用高

性能的图形工作站。

总控系统中的各项具体测控仿真功能，由各类测控板卡具体实现。测控板卡实现的具体功能与技术要求一一对应，达到硬件功能与逻辑功能的统一。总控系统软件采用 VC++ 语言进行开发，遵循模块化原则，构建友好的、功能完备的测试界面，便于操作人员使用、调试开发与维护。

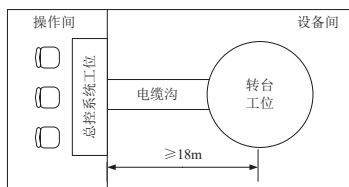
(3) 总体设计概述

总控系统的总体硬件主要包括：

- ① 主控计算机系统 CTS
- ② 遥测计算机系统 RTS
- ③ 实时视景计算系统 ITS
- ④ 电源单元 PDU

主控测试计算机系统 CTS、遥测计算机系统 RTS 和实时视景计算系统 ITS 通过星型 VMIC 组成一个实时计算机网络平台。而主控计算机系统 CTS 亦可通过 GPIB 总线和 RS232 接口对电源单元 PDU 进行程控。

由于被测产品位于运动转台之上，而总控系统距离转台距离比较远（ $\geq 18\text{m}$ ，如下图所示），同时转台上除产品安装位置外可供安装其它设备（模块）的空间区域非常有限。



总控系统安装布置空间简图

因此，总控制系统与被测产品的信号连接多数要直接经过长线传输；另一部分通过在前端（即转台上靠近产品的位置）进行处理后再与被测产品进行互联。主控计算机系统 CTS 中的 1553B、ARINC429、RS422 等总线信号以及一次性指令的数字量输入 / 输出信号（即 DI/DO 信号）和 V/I 注入信号，可进行长线传输与被测产品直接互联。主控计算机系统 CTS 中的脉冲注入信号属于高速脉冲信号，需要主控调理模块将其变换为差分信号形式后进行长线传输，至转台后由前端调理系统接收，隔离、驱动后变换为 TTL 信号后再近距离发送给被测产品。遥测计算机系统 RTS 中采集的被

虚拟仪器 3.0 的应用领域

测产品发出的数字遥测 BMK 信号，属于并行信号，需要在前端调理系统进行隔离，并变换为差分信号形式，再传输至遥测计算机系统中的遥测调理模块，由其再变换为标准并行 BMK 信号送入遥测计算机系统采集。

为了确保更为可靠地接收来自被测产品的 LVDS 数据，遥测计算机系统 RTS 中采集的 LVDS 信号在前端调理系统里进行了缓冲和驱动。实时视景计算机系统 ITS 主要用于仿真过程中的动态图形显示以及仿真后的显示输出与动画回放，不涉及与被测产品的信号互连。因此，经处理后需要变换为差分信号形式进行长线传输的信号归总如下：

① 数字遥测 BMK 信号

② 脉冲注入信号

经长线与被测产品直接互连的信号归总如下：

① 1553B 总线信号

② LVDS 信号（经缓冲和驱动）

③ ARINC-429 总线信号

④ RS422 总线信号

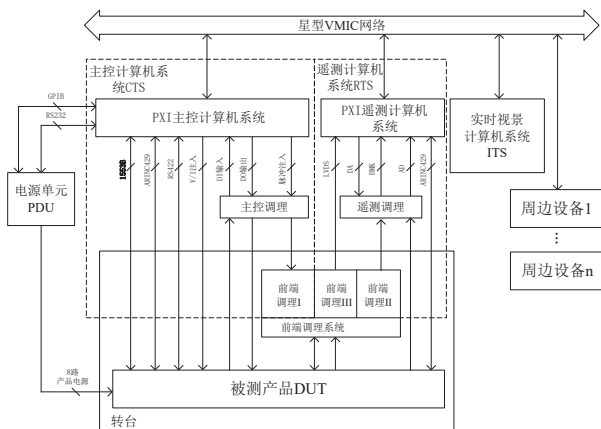
⑤ V/I 注入信号

⑥ 一次性指令 DI/DO 信号

⑦ A/D、D/A 信号

⑧ 产品电源供电

总控系统的总体逻辑功能结构如下图所示。



总体逻辑功能结构图

从图中可以明确看出，主控计算机系统 CTS 包括 PXI 主控计算机系统、主控调理以及位于前端调理系统中的前端调理 I。主控调理 I 将 TTL 电平的脉冲注入信号变换为差分信号进行高速、长距离传输，至前端调理 I 再转换为 TTL 电平脉冲信号，最终注入到被测产品。同时，主控调理 II 还要对一次性指令信号（数字量 DI/DO）进行隔离、驱动。主控调理 III 则将 V/I 注入卡输出的三路同步电压信号变换为三路隔离的同步电流信号注入到产品。

遥测计算机系统 RTS 包括 PXI 遥测计算机系统、遥测调理以及位于前端调理系统中的前端调理 II 和前端调理 III。前端调理 II 将并行数字遥测 BMK 信号变换为差分信号进行高速、长距离传输，至遥测调理后变换为数字遥测 BMK 信号，最终由 PXI 遥测计算机系统 BMK 接收卡进行采集和存储。前端调理 III 对发自产品的 LVDS 进行缓冲、驱动及预处理后再传送到遥测计算机进行采集、存储等处理。同时，遥测调理将模拟遥测信号进行隔离、变换后进入遥测计算机系统 RTS 的 AD 采集通道进行采集、分析与存储。

前端调理系统紧凑、近距离安装在被测产品附近工位。

(4) 主控计算机系统 CTS

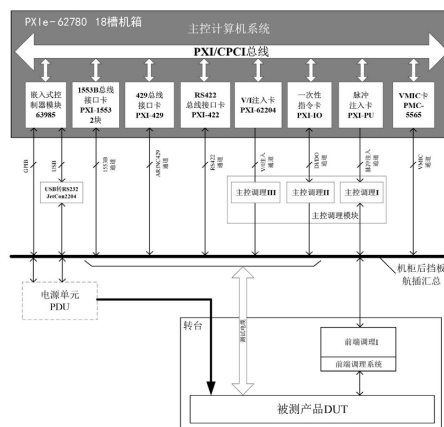
主控计算机系统 CTS 是整个平台的核心构成部分，其主要包括以下组成部分：

① PXI 主控计算机系统

② 主控调理模块

③ 前端调理 I

主控计算机系统 CTS 结构简图如下图所示。



主控计算机系统结构简图

虚拟仪器 3.0 的应用领域

主控计算机系统 CTS 中的主控调理模块又包括了主控调理 I、主控调理 II 和主控调理 III 三个组成部分。主控计算机系统 CTS 中的前端调理 I 近距离、紧凑安装于被测产品附近工位。

(1) PXI 主控计算机系统

PXI 主控计算机系统以简仪公司的 PXI 嵌入式控制器和 PXI 机箱为核心，配置必要的 PXI 接口测控板卡进行构建，其具体配置如下：

- ① PXI 机箱：简仪公司 18 槽 PXI 机箱 PXIe-62780
- ② PXI 嵌入式控制器：简仪公司 PXIe-63985
- ③ 1553B 总线接口卡：PXI-1553 件数：2 块
- ④ ARINC429 总线接口卡：PXI-429 件数：1 块
- ⑤ RS422 接口卡：PXI-422 件数：1 块
- ⑥ V/I 注入卡：PXI-62204 件数：1 块
- ⑦ 脉冲注入卡：PXI-PU 件数：1 块
- ⑧ 一次性指令卡：PXI-IO 件数：1 块
- ⑨ VMIC 卡：PMC-5565 件数：1 块
- ⑩ VMIC 集线器：ACC-5595 件数：2 台
- ⑪ USB 转 RS232 模块：JetCon2204 件数：1 块

(2) 主控调理模块

主控调理模块是主控计算机系统的重要组成部分，由两个子模块组成，配置如下：

- ① 主控调理 I（脉冲注入调理）：CS-1 件数：1 块
- ② 主控调理 II（数字量输入 / 输出调理）：CS-2 件数：1 块
- ③ 主控调理 III（V/I 注入调理）：CS-3 件数：1 块

(3) 前端调理 I

前端调理 I 主要用于将主控调理模块中的主控调理 I 发出的、经长线传输的差分脉冲信号转换为单端 TTL 信号后，再转发给被测产品。其配置如下：

- ① 前端调理 I：FS-1 件数：1 块

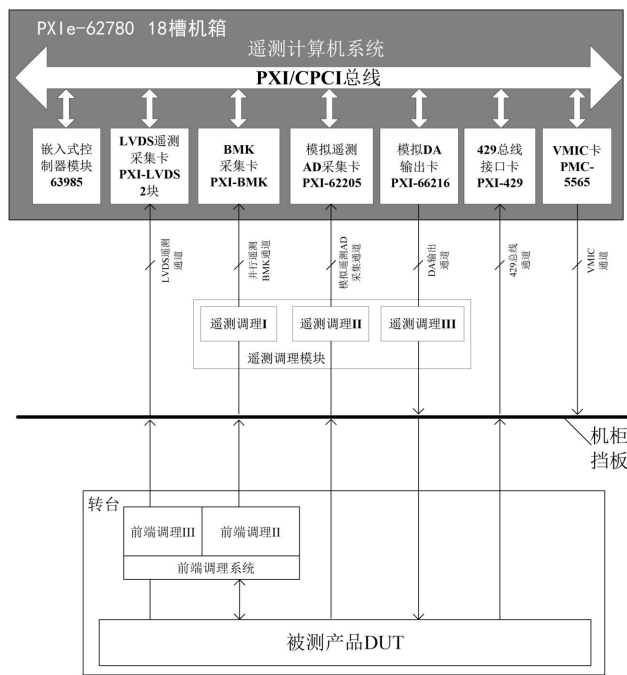
(5) 遥测计算机系统 RTS

遥测计算机系统也是总控系统的重要组成部分，其主要完成被测产品 LVDS 遥测数据、并行遥测 BMK 数据、模拟遥测数据的实时采集、解析和保存与分析等功能。同时本部分还配置有模拟量 DA 输出卡，用于测试设备的自检与校准等功能。

遥测计算机系统 RTS 主要包括以下组成部分：

- ① PXI 遥测计算机系统
- ② 遥测调理模块
- ③ 前端调理 II
- ④ 前端调理 III

遥测计算机系统 RTS 结构简图如下图所示。



遥测计算机系统结构简图

(1) PXI 遥测计算机系统

PXI 遥测计算机系统以简仪公司的 PXI 嵌入式控制器和 PXI 机箱为核心，配置必要的 PXI 接口测控板卡进行构建，其具体配置如下：

- ① PXI 机箱：简仪公司 18 槽 PXI 机箱 PXIe-62780

虚拟仪器 3.0 的应用领域

- ② PXI 嵌入式控制器：简仪公司 PXIe-63985
- ③ LVDS 遥测采集卡：PXI-LVDS 件数：2 块
- ④ BMK 采集卡：PXI-BMK 件数：1 块
- ⑤ 模拟遥测 AD 采集卡：PXI-62205 件数：1 块
- ⑥ 模拟 DA 输出卡：PXI-66216 件数：1 块
- ⑦ ARINC429 总线接口卡：PXI-429 件数：1 块
- ⑧ VMIC 卡：PMC-5565 件数：1 块

(2) 遥测调理模块

遥测调理模块是遥测计算机系统的重要组成部分，其由两个子模块组成，具体配置如下：

- ① 遥测调理 I（遥测 BMK 调理）：RS-1 件数：1 块
- ② 遥测调理 II（模拟遥测调理）：RS-2 件数：1 块
- ③ 遥测调理 III（DA 输出调理）：RS-3 件数：1 块

(3) 前端调理 II

前端调理 II 主要用于将被测产品发出的并行遥测 BMK 信号转换为差分信号进行长线传输至遥测调理 I，而后由遥测调理 I 变换为数字遥测 BMK 信号送入 PXI 遥测计算机系统。其配置如下：

- ① 前端调理 II：FS-2 件数：1 块

(4) 前端调理 III

前端调理 III 主要用于将被测产品发出的 LVDS 信号进行缓冲和驱动后再转发给 LVDS 遥测采集卡。其配置如下：

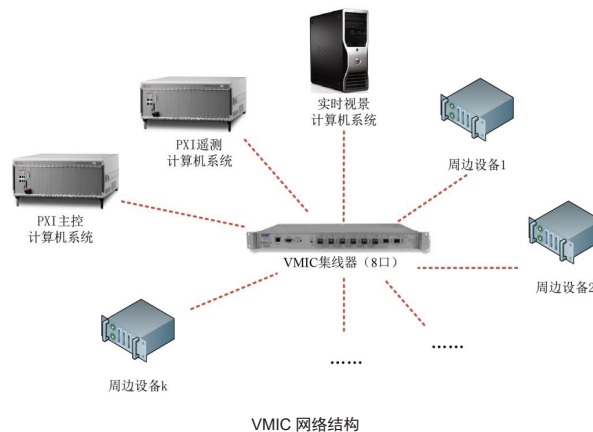
- ① 前端调理 III：FS-3 件数：1 块

VMIC 网络

总控系统内部的三个计算机系统，及其与外部的其它周边计算机系统通过 VMIC 网络（实时反射内存网）组成一个实时、网络化计算机测试仿真平台。

根据技术要求，总控制系统的 VMIC 网络利用集线器互联成星型网络结构（如所示），且可以配置为多模光

纤接口与单模光纤接口的任意组合。一个集线器可连接 8 个反射内存节点。系统运行时，不需要每个节点上电运行。且集线器能够自动隔离故障节点。



技术难点及解决方案

(1) 概述

方案中对于信号的长线传输给予了高度重视和精心的设计，力求简洁、合理。方案中将总控系统与被测产品的互连的信号进行细致分类，充分利用了信号本身的特点和特性以及设备工作的实际状况。

经长线与被测产品直接互连的信号归总如下：

- ① 1553B 总线信号
- ② LVDS 信号（经缓冲和驱动）
- ③ ARINC-429 总线信号
- ④ RS422 总线信号
- ⑤ V/I 注入信号
- ⑥ 一次性指令 DI/DO 信号
- ⑦ A/D、D/A 信号
- ⑧ 产品电源供电

经处理后需要变换为 RS422 差分信号形式进行长线传输的信号归总如下：

- ① 数字遥测 BMK 信号

虚拟仪器 3.0 的应用领域

② 脉冲注入信号

(2) 分类分析

总控系统涉及到的 1553B、ARINC429、RS422 信号，其均为较低速率的总线信号，传输时均为差分信号，其信号本身特性与特点就决定了可以进行长线传输。因此，方案中对这些信号均未进行变换，与被测产品经长线互连。当然，这些总线信号在进行较长距离传输时，要充分注意构建合理的总线结构，合理终结，选择优质的传输介质。

总控系统涉及到的一次性指令实质属于数字量信号的输入/输出。由于其传输速率比较低，而且方案中数字量输入/输出均为经过了隔离、驱动等调理，因此经长线传输后与被测产品直接互连。

总控系统涉及到的 LVDS 遥测信号，由于 LVDS 信号本身的特性，可以进行较高速率的长线传输。因此，方案中将被测产品发出的 LVDS 遥测信号直接进行长线传输至总控系统，进行采集。

总控系统涉及到的 V/I 注入信号为三路，其本质是三个高精度的程控电流源。而对于电流信号来说，可以适应长线传输。

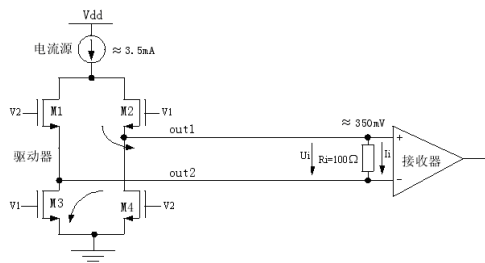
总控系统涉及到的 AD 模拟量信号是来自于被测产品的、变换速率不高的模拟信号，因此可以由被测产品经长线传输后至总控系统。

总控系统中的产品供电均为高精度程控电源，而产品对供电电压误差范围较宽，因此产品供电可以直接进行长线传输与被测产品互连。如需更高精度，则每路供电电源可以采用四线制，即远端采样来提高产品的供电精度。

总控系统涉及到的数字遥测 BMK 信号以及脉冲注入信号，则不适于长线传输，必须进行必要的变换处理。方案中将其均变换为 LVDS 信号形式进行长线传输。

(3) LVDS 遥测信号

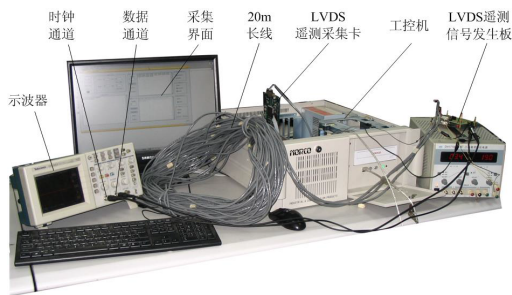
来自产品的 LVDS 遥测信号包括一路 LVDS 信号形式的时钟，一路 LVDS 信号形式的数据。它们的长线传输问题，实际上就是 LVDS 信号本身的长线传输能力。LVDS 是 Low Voltage Differential Signaling，是一种低电压摆幅的差分信号技术。其具有高速、低功耗、低噪声等突出特点。其典型的收发工作原理如下图所示。



LVDS 的工作原理示意图

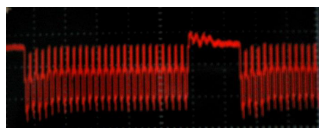
从图中可以看出其发送器是一个典型的电流源，而接收器是一个典型的比较器。因此，其天然的具有高速、长距离传输的能力。National Semiconductor 即国家半导体公司在第四版的《LVDS Owner's Manual》中，对 LVDS 信号的长线传输也进行了阐述。《LVDS Owner's Manual》中列举了不少 LVDS 信号长线传输的实例。例如：高速（1.65Gbps）DVI 信号（含 6 路数据、1 路时钟、3 路控制信号）可以稳定、可靠传输 5~30m。又例如：高速（1.65Gbps）HDMI 信号（含 3 路数据、1 路时钟、3 路控制信号）也可以稳定、可靠传输 5~30m。而被测产品发出 LVDS 遥测信号仅包含 1 路时钟和 1 路数据，时钟速率典型值为 33MHz，因此可以进行 > 18m 距离的长线传输。

同时，北航自动测试技术实验室也对 LVDS 遥测信号的长线传输问题进行了初步研究。对符合 LVDS 信号标准的遥测数据（含 1 路 33MHz 时钟，1 路数据）进行了 20m（五类双绞线）左右的传输，而后利用 LVDS 遥测采集卡进行采集、存储（实验场景如下图所示）。实测波形良好、采集稳定、可靠，暂未出现丢帧、漏帧、无法采集等错误，信号延时约 100ns 左右。也就是说，实验表明 LVDS 遥测信号可以 > 18m 距离的长线传输。为了确保更为可靠地接收来自被测产品的 LVDS 数据，遥测计算机系统 RTS 中采集的 LVDS 信号在前端调理 III 中进行了缓冲和驱动。

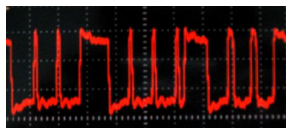


LVDS 遥测数据长线实验场景简图

虚拟仪器 3.0 的应用领域



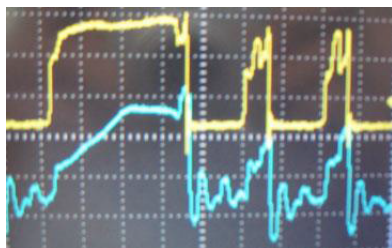
LVDS 遥测时钟信号



LVDS 遥测数据信号

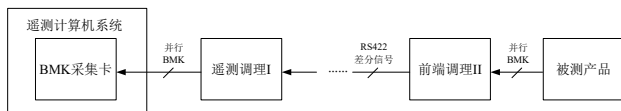
(4) 遥测 BMK 信号

遥测 BMK 信号属于高速、并行信号，其不适于进行长距离传输。例如，实验将 BMK 采集卡发出的 BMK 信号利用 20m 五类双绞线传输至 BMK 采集卡的采集通道，进行自发自收的自检实验，并测量典型信号的波形。实验表明，经 20m 双绞线传输的 BMK 信号无法进行自检，同时相关信号经长线传输后发生严重的波形畸变。所以，实验表明遥测 BMK 信号无法直接进行 20m 左右的长线传输。



BMK 直接长线传输后波形

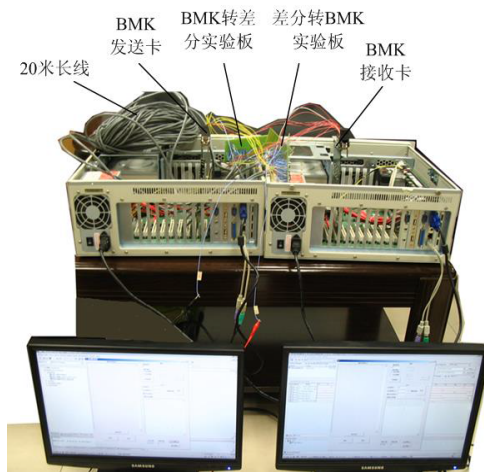
在方案中，提出了如下解决策略：将产品发出的并行数字遥测 BMK 信号先经前端调理 II 转换为 RS422 差分信号形式，而后进行长线传输。传输至总控系统后，经遥测调理 I 再转换为标准 BMK 信号，最终进入遥测计算机系统。前端调理 II 可以利用目前转台已有的工位，与被测产品近距离安装。其简要工作过程如错误！未找到引用源。所示。



BMK 长线传输工作过程

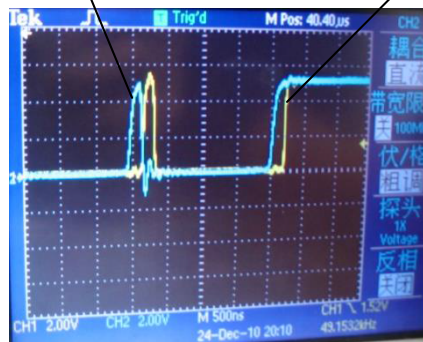
为了验证这种方法策略的可行性，北航自动测试技术实验室对 BMK 信号变换为 RS422 差分信号形式后的长线传输问题进行了深入研究。将 BMK 采集卡发出的 BMK 信号经实验板变换为 RS422 差分信号后，经 20m 五类双绞线传输再回送至实验板，由实验再将 RS422 差分信号变换为 BMK 后，最终送至 BMK 采集卡的采集通道，进行自发自收的自检实验，并测量典

型信号的波形。实验场景如下图所示所示。实验进行了多次遥测 BMK（帧帧间隔 12ms，每帧 256 个 BMK 字）的发送、采集以及存储实验，未发生丢帧、漏帧、误码等错误，实测波形良好（如下图所示），延时约 200ns。也就是说，实验表明这种策略是完全可行的。



遥测 BMK 转 RS422 差分信号长线实验场景简图

原始 EN 信号 经 RS422 差分长线传输后 EN 信号



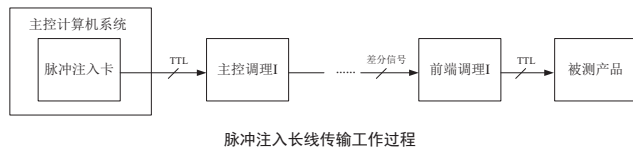
EN 信号实测波形

BMK 转 RS422 差分信号实验结果简表			
编号	实验次数 单位：万次	实验时间 单位：分钟	结果
1	15.0	30.0	无错误
2	10.0	20.0	无错误
3	15.0	30.0	无错误
4	15.0	30.0	无错误
5	30.0	60.0	无错误
6	23.0	43.0	无错误
7	35.0	70.0	无错误
8	16.0	32.0	无错误
9	52.0	104.0	无错误
10	47.0	94.0	无错误

虚拟仪器 3.0 的应用领域

(5) 脉冲注入信号

技术指标要求脉冲注入信号需要经过 18m 传输后再注入被测产品。但是脉冲注入卡输出的 TTL 脉冲信号脉宽小 (μs 级)、频率高 (亚兆 Hz 级), 显然不适于进行长线传输。因此方案中将脉冲注入卡输出的各个通道脉冲信号经主控调理 I 变换为 RS422 差分信号后, 再进行长线传输。一直传输到达产品附近的前端调理 I, 前端调理 I 将 RS422 差分信号反变换为单端 TTL 脉冲信号后, 再转发给被测产品。其简要工作过程如下图所示。



脉冲注入信号的频率与遥测 BMK 的频率接近。因此, 既然遥测 BMK 信号可以转换为 RS422 差分信号进行尽 20m 的长线传输, 那么脉冲注入信号变换为 RS422 差分信号进行同样距离的长线 (五类双绞线) 传输也是完全可行的。

飞控电子组合测试系统

(1) 概述

飞控电子组合测试系统主要由软件、控制系统、信号调理单元、电源、专用适配器、高低温箱等组成, 可同时测试 5 套被测产品。飞控电子组合测试系统主要用于实现如下测试功能:

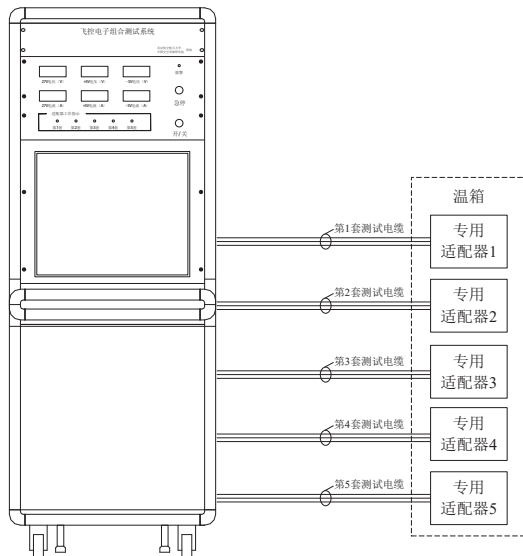
- ① 惯测接口部件的常温和温度测试;
- ② 总线接口部件的常温和温度测试;
- ③ 译码控制部件的常温和温度测试;
- ④ 飞控电子组合常温测试。

其中, 温度测试时低温 -50°C 、高温 75°C , 温变率 $5^{\circ}\text{C}/\text{min}$, 循环次数 15 次。

(2) 系统组成与工作原理

飞控电子组合测试系统由主控台和五套专用适配器组成。主控台由工控机、各种测控板卡、信号调理单元、

信号切换分配单元、电源单元、自检软件及综合测试软件组成, 主控台的所有硬件设备都位于标准单联机柜内。五套测试适配工装 (可放入温箱) 通过五套电缆与主控台相连, 其结构组成如下图所示。



飞控电子组合测试系统简图

(3) 总体设计概述

针对技术要求, 飞控电子组合测试系统的总体硬件主要包括:

- ① 测试计算机系统 CTS;
- ② 信号调理 SCU;
- ③ 电源单元 PDU;
- ④ 信号切换分配单元 SDU;
- ⑤ 专用适配器 TAT

其中, 最为核心的测试计算机系统 CTS 以 CPCI 工控计算机为核心构建, 其涵盖了控制器和各种测控板卡, 测试台通过 CPCI 总线的测控板卡实现如下功能:

- ① 一次性指令的发送和接收
- ② BMK 信号的接收
- ③ 429 总线信号的发送和接收
- ④ 陀螺和加速度信号的发送
- ⑤ 温度及修正常数读取

虚拟仪器 3.0 的应用领域

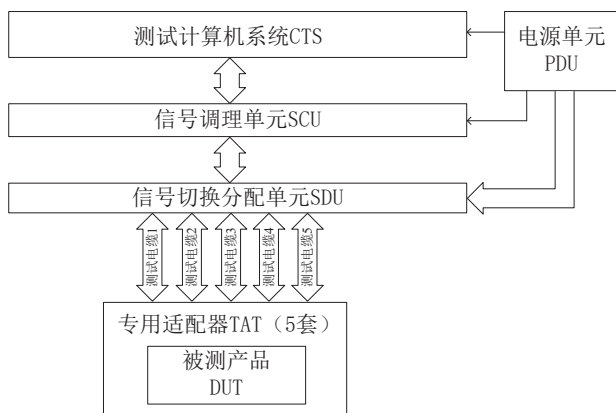
⑥ RS422 接口

⑦ 导引头双口 RAM 读写模拟

⑧ 计算机板接口模拟

⑨ 电源控制

上述各种测控板卡的输出信号经过信号调理单元 SCU 的调理再经过信号切换分配单元 SDU 的分配后, 形成 5 套电子组合测试电缆 1、2、3、4、5 后, 最后通过专用适配器与被测产品 DUT 互连, 通过综合测试软件实现各种测试功能。飞控电子组合测试系统总体结构如错误! 未找到引用源。所示。



飞控电子组合测试系统总体结构原理图

(4) 测试计算机系统 CTS

飞控电子组合测试系统中的测试计算机系统 CTS 由基于 CPCI 总线的工控计算机和 CPCI 总线接口的 7 块测控板卡构成, 其具体构建如下所述。

① CPCI 机箱: 简仪公司 18 槽 PXI 机箱

② 嵌入式控制器: 简仪公司的 PXI 嵌入式控制器

③ CPCI 总线测控板卡 测试设备的核心组成部分, 主要包括以下组成部分:

◆ 一次性指令控制 / 导引头通讯模拟卡: CPCI-IO/RAM

◆ BMK 接收卡: CPCI-BMK

◆ ARINC429 总线接口卡: CPCI-429

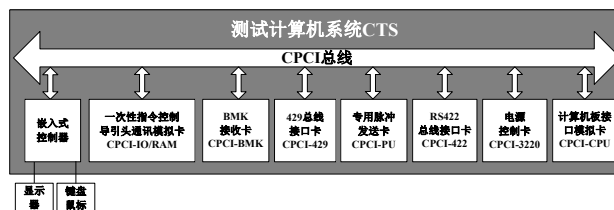
◆ 专用脉冲发送卡: CPCI-PU

◆ RS422 总线接口卡: CPCI-422

◆ 电源控制卡: CPCI-3220

◆ 计算机板接口模拟卡: CPCI-CPU

飞控电子组合测试系统测试计算机系统 CTS 详细的总体构建结构如下图所示。



测试计算机系统结构图

(5) 信号调理单元 SCU

信号调理单元 SCU 主要功能如下:

◆ 对一次性指令发送和接收卡的输入 / 输出数字量信号进行必要的隔离和驱动。

◆ 对导引头通讯模拟卡输入 / 输出数字量信号进行必要的隔离和驱动。

◆ 对计算机板接口模拟卡输入 / 输出数字量信号进行必要的隔离和驱动。

◆ 对电源单元 PDU 给产品的 3 路电压量和相应供电电流进行必要的变换, 以便形成标准范围内 (典型值: 0 ~ 10V) 的模拟量, 进入电源控制卡。

◆ 对被测产品的 5 路 27V 信号进行必要的衰减变换, 以便形成标准范围内 (典型值: 0 ~ 10V) 的模拟量, 进入电源控制卡后进行采集。

因此其主要包括以下组成部分:

◆ 一次性指令输入 / 输出调理 SCB1-1

◆ 导引头通讯模拟卡输入 / 输出调理 SCB1-2

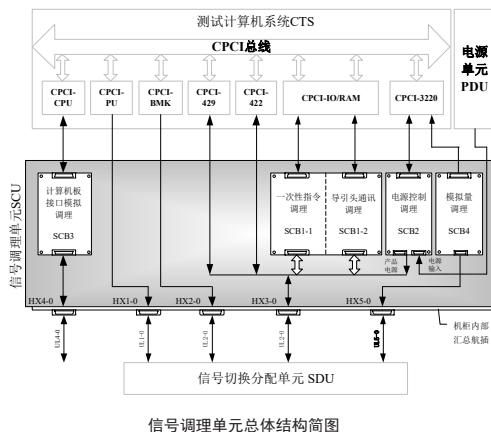
◆ 电源控制调理 SCB2

◆ 计算机板接口模拟卡输入 / 输出调理 SCB3

◆ 模拟量输入调理 SCB4

虚拟仪器 3.0 的应用领域

信号调理单元 SCU 总体结构如下图所示。



- ◆ 对遥测信号 (UL2-0 电缆) 进行 1 分 5 切换分配控制;
- ◆ 对工作信号 (UL3-0 电缆中的非电源信号) 进行 1 分 5 切换分配控制;
- ◆ 对工作信号 (UL3-0 电缆中的电源信号) 进行 1 分 5 切换分配控制。

针对上述功能, 其主要包括以下组成部分:

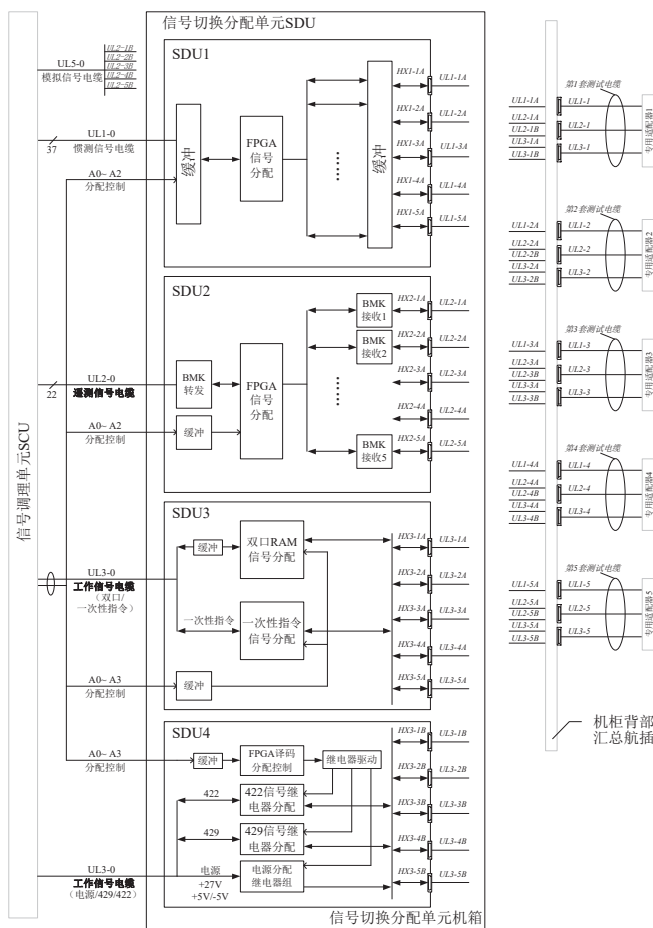
- ◆ 惯测信号 (UL1-0 电缆) 1 分 5 切换分配控制 SDU1
- ◆ 遥测信号 (UL2-0 电缆) 1 分 5 切换分配控制 SDU2
- ◆ 工作信号 (UL3-0 电缆中的双口 RAM 和一次性指令信号) 1 分 5 切换分配控制 SDU3
- ◆ 工作信号 (UL3-0 电缆中的电源和 429 及 422 信号) 1 分 5 切换分配控制 SDU4

(6) 信号切换分配单元 SDU

信号切换分配单元 SDU 主要功能如下:

- ◆ 对惯测信号 (UL1-0 电缆) 进行 1 分 5 切换分配控制;

信号切换分配单元 SDU 总体结构如下图所示。



信号切换分配单元总体结构简图

虚拟仪器 3.0 的应用领域

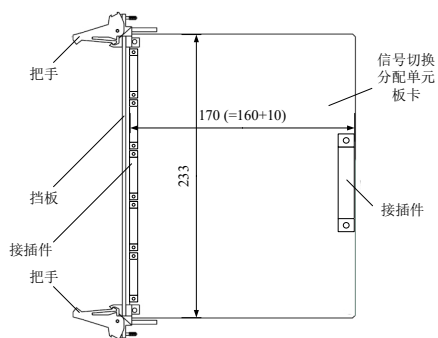
(7) 信号切换分配单元设计

信号切换分配单元 SDU 是系统的重要组成部分，其包括以下四个组成部分：

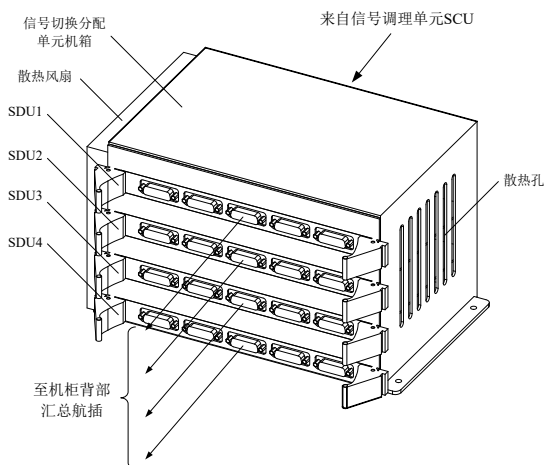
- ◆ 惯测信号（UL1-0 电缆）1 分 5 切换分配控制 SDU1
- ◆ 遥测信号（UL2-0 电缆）1 分 5 切换分配控制 SDU2
- ◆ 工作信号（UL3-0 电缆中的双口 RAM 和一次性指令信号）1 分 5 切换分配控制 SDU3
- ◆ 工作信号（UL3-0 电缆中的电源和 429 及 422 信号）进行 1 分 5 切换分配控制 SDU4

综合考虑信号切换分配单元的安装布置、调试、维护，将这三块调理板设计为标准的 6U 标准板卡形式，安装于一个信号切换分配单元机箱中。同时，三块板卡采用抽拉的安装方式，以求便于快速安装、更换。

6U 标准板卡（233X170）的具体外形尺寸如下图所示。



6U 信号切换分配单元板卡结构示意图



信号切换分配单元机箱示意图



信号切换分配单元机箱实物简图

飞控电子组合测试系统

(1) 概述

导引头综合测试系统用于在常温状态下对（红外成像）导引头的主要功能进行定量检测。该测试系统可以实现以下主要测试项目及指标：

① 导引头控制系统参数标定

- ◆ 陀螺零位标定；
- ◆ 旋变零位、粗精级相位差标定；
- ◆ 电位计零位标定；
- ◆ 电位计比例尺标定；
- ◆ 陀螺比例尺标定；
- ◆ 陀螺信号非正交耦合补偿系数标定；
- ◆ 内环稳定回路的增益系数。

② 导引头控制系统参数测试

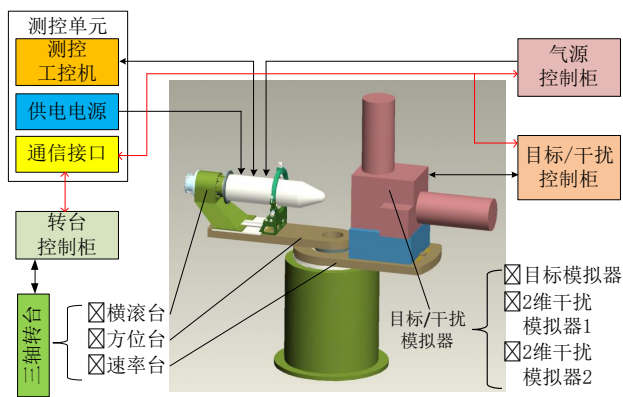
- ◆ 陀螺零位误差和锁定噪声测试；
- ◆ 电位计零位测试；
- ◆ 电位计比例尺测试；
- ◆ 陀螺比例尺测试；
- ◆ 旋变零位测试；
- ◆ 陀螺非正交解耦测试；
- ◆ 位置回路测试；

虚拟仪器 3.0 的应用领域

- ◆ 稳定回路测试；
 - ◆ 跟踪回路测试；
 - ◆ 动态锁定品质测试；
 - ◆ 扫描场测试；
 - ◆ 随动精度测试；
 - ◆ 跟踪场测试；
 - ◆ 随动时间测试；
 - ◆ 跟踪能力测试。
- ③ 导引头性能测试
- ◆ 对外接口对地电阻测试；
 - ◆ 电源电压、消耗电流测试；
 - ◆ 电锁精度测试；
 - ◆ 电锁品质测试；
 - ◆ 导引精度测试；
 - ◆ 导引对称性测试；
 - ◆ 导引耦合系数测试；
 - ◆ 视场测试；
 - ◆ 捕获场测试；
 - ◆ 跟踪场测试；
 - ◆ 制冷时间测试；
 - ◆ 蓄冷时间测试；
 - ◆ 制冷维持流量测试；
 - ◆ 随动性能测试；
 - ◆ 跟踪回路超调量测试；
 - ◆ 抗人工干扰测试；
 - ◆ 成像质量测试；
 - ◆ 盲元特性测试。

(2) 总体设计

该测试系统由测控单元、红外目标 / 干扰模拟器、三轴转台、供气单元、装夹装置等部分组成，系统详细组成框图如图所示。



测试系统总体结构示意图

① 测控单元

测控单元包括以下组成部分

◆ PXI 主控计算机

- a. PXI 控制器 型号：PXIe-63985（简仪）
- b. PXI 机箱 型号：PXIe-62780（简仪）
- c. 红外图像采集卡 型号：PXI-3116（自研）
- d. 多功能 AD 采集卡 型号：PXI-3220（自研）
- e. RS422 通讯卡 型号：PXI-3440（自研）
- f. 台式万用表 型号：34401A（Agilent）
- g. 外置磁盘阵列 型号：DS1512

② 调理单元

调理单元主要包括以下组成部分：

- ◆ 图像调理电路模块
- ◆ 多功能 AD 采集调理
- ◆ 开关矩阵电路模块
- ◆ 电源监控调理模块
- ◆ SPI 烧写电路

虚拟仪器 3.0 的应用领域

◆ 双口 RAM 转 RS422 模块

③ 电源单元

根据被测导引头技术要求，为了保证产品供电电源的品质、可靠性和操控的方便性，方案选用台湾顾纬公司生产的可编程直流线性稳压电源 PSM - 6003 和 PPE-3323。

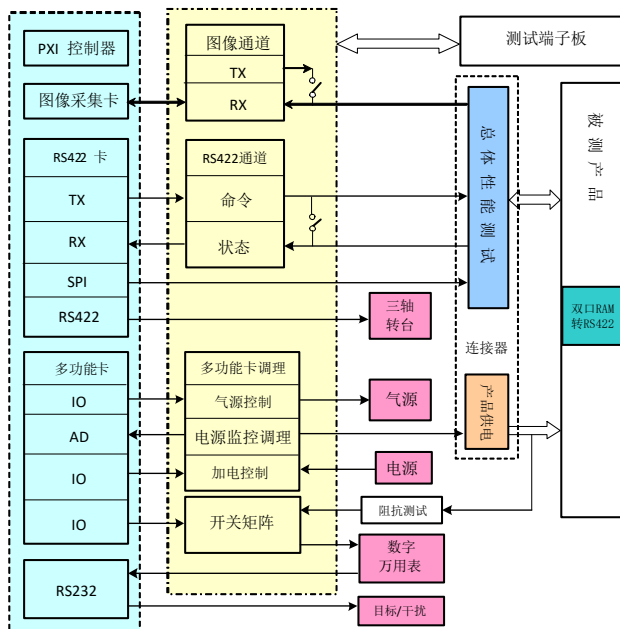
◆ PSM - 6003

直流输出 0 ~ +30V / 6A
 0 ~ +60V / 3.3A

◆ PPE-3323

电压 0 ~ +32V, 0 ~ -32V, 3.3V/5V 固定
电流 0 ~ +3A, 0 ~ -3A, 3A 固定

测控单元硬件连接关系如下图所示。PXI 主控计算机的测试卡通过调理电路后形成了可以和产品对接的信号，主要包括图像、422、AD、IO 和电源。将这些信号连接到机柜后面板的几个航插（如产品供电、性能测试、气源控制和转台控制等）上，通过航空插头使测试设备和被测产品可以快速、准确、无误地连接。



测控单元硬件连接图

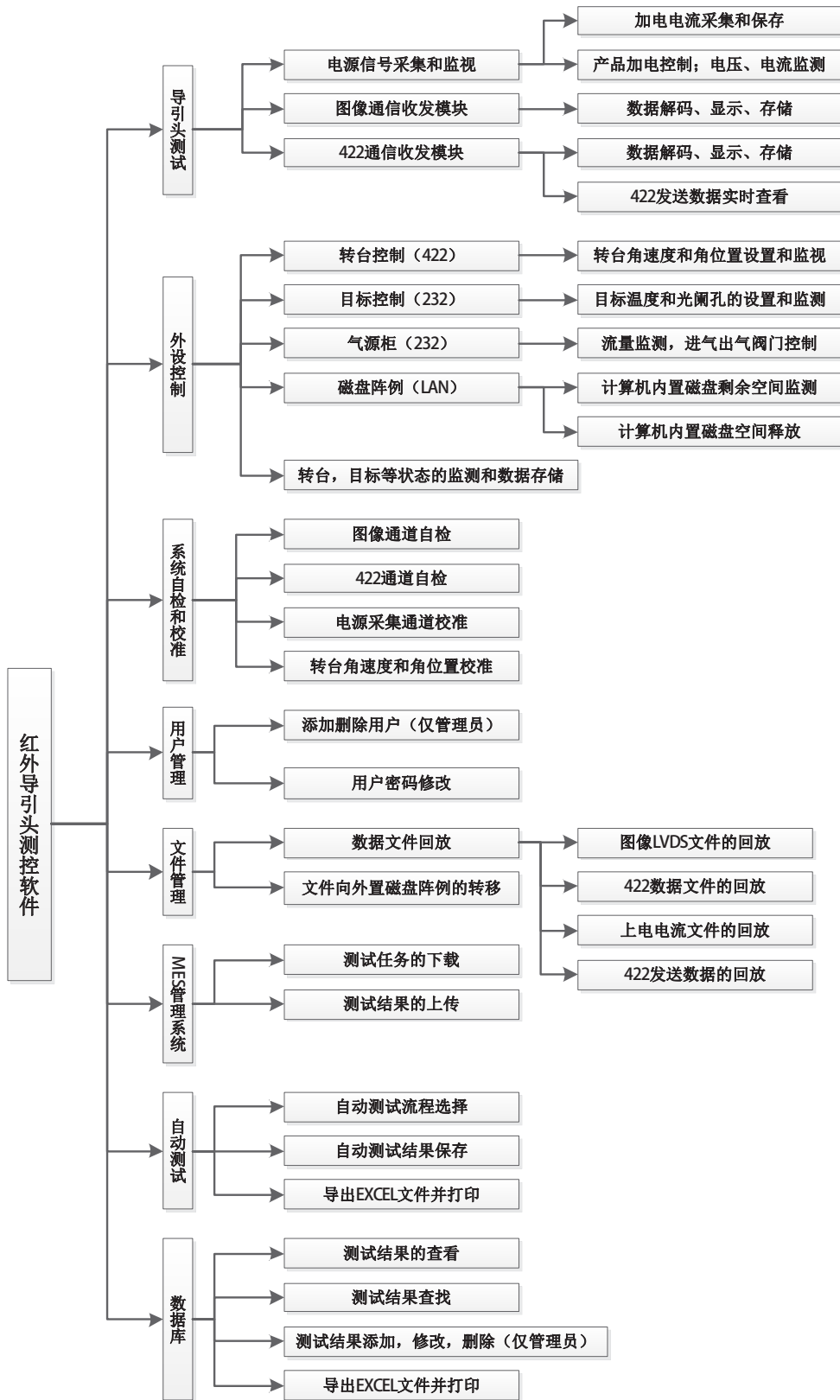
(3) 测试软件

① 测试软件总体设计

测试系统采用了效率较高的 C# 语言进行开发。整个测试软件运行于 WindowsXP 操作系统，编译环境选用 Visual studio 2005。为了进一步缩短软件开发周期，提高软件的可靠性，测试软件配合使用了简仪公司的 seesharp tools 测控类插件。

测试软件主要实现导引头红外图像、LVDS 数据及 422 数据的实时采集、解算、显示、存储、回放、数据分析、后期处理等功能。同时能实现测试设备的自检测和校准。软件设计将采用模块化结构，整个测控软件划分为以下几个基本模块：导引头测试模块、外设控制，系统自检和校准、用户管理、文件管理，MES 管理系统，自动测试，测试结果数据库，数据分析和处理模块。其总体组成方式如下图所示。

虚拟仪器 3.0 的应用领域

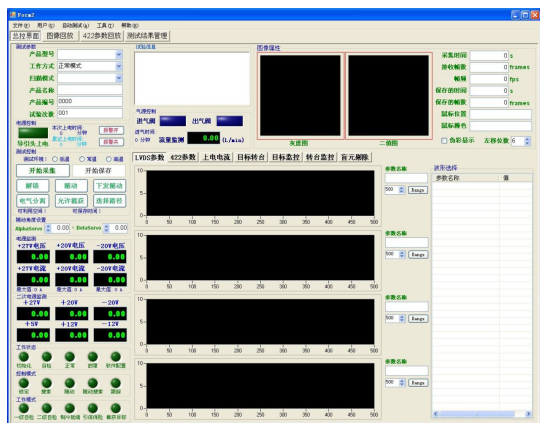


测控系统软件总体组成结构简图

虚拟仪器 3.0 的应用领域

② 测试界面设计

测试程序主界面如下图所示。



程序主界面

测试程序主界面主显示的项目如下所述：

◆ 产品信息

产品信息部分的主要功能是设置和查看当前正在测试的产品的的相关信息，如：产品编号，试验次数，产品名称，工作方式等。

◆ 电源控制和监测

主要包括：电源加电，断电，一次电源电压、电流监测数据，二次电源电压监测和显示，同时还根据产品编号而保存的产品总供加电时间，本次加电时间等。

◆ 气源控制和监测

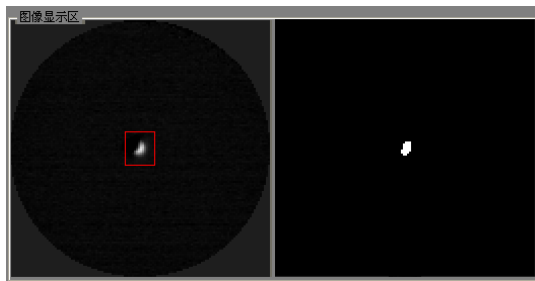
主要有气源供气时间，进气，出气阀通断状态，氮气流量监测和显示。

◆ 转台控制和监测

包括：转台加电、断电、停止等功能。还有转台的实时角位置和角速度显示，同时还可以设置转台的运动方式。

◆ 图像及其属性显示

主要有图像的灰度图显示和二值图显示，当前帧数的显示等。其中，当产品捕获到目标时，会自动用红色矩形框标出目标所在位置。下图演示的就是捕获到目标后，测试界面上的显示效果。



捕获目标时图像显示状态

此外，可以用鼠标在图像显示窗口中任意拖拽，实现图像放大或平移的功能。系统还可根据鼠标所在位置，显示出当前点的灰度及位置信息。

◆ LVDS 和 422 参数查看

LVDS 参数波形显示的内容为用户从参数列表中选择出来的比较关注的导引头状态信息。以曲线的方式表现出来，使用户能非常直观地观察到数据的变化趋势。但由于实时绘制参数曲线十分消耗系统资源，同时绘制过多曲线会使图像采集的实时性受到影响。因此，只能同时显示少量几个参数，由用户自行挑选。

422 参数显示页面包括坐标导航图、422 参数波形显示和 422 参数列表显示三部分。其中，坐标导航图是将位标器以及雷达的测试信号中包含相位信息，显示为坐标导航图的形式。

◆ 导引头工作状态监视

对导引头的工作状态以 LED 的方式直观地显示。

◆ 目标控制和监视

主要包括：目标的温度，光阑，快门等参数的设置，和目标实时的状态显示。

◆ 导引头指令下发

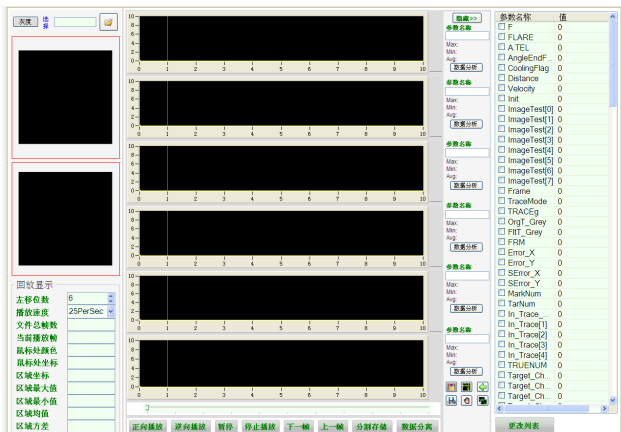
包括常用的导引头下发指令，如锁定、解锁、下发随动等。

③ 图像回放设计

图像回放包括图像灰度和二值图的显示；当前回放信息的显示；当前选中参数波形的显示和所有可用参数的列表；还有当前选中段部分的计算值，如最大、最小、平均值等。

虚拟仪器 3.0 的应用领域

图像回放界面如下图所示。



图像回放页面

图像回放界面有导出数据功能，可以将当前的一帧图像数据以 16 进制的形式导出到 EXCEL 表中，以供测试者查看和分析计算；图像回放数据可以在选择并拖动波形时，实时计算和显示计算的值；还有导出当前波形全部数据或部分数据的实用功能。

虚拟仪器 3.0

开源 面向对象 跨平台



网址：www.jytek.com
电话：021-50475899
邮箱：info@jytek.com