摘　要

地铁信号设备中输入输出设备是信号逻辑和现场设备之间的接口，有着四高（高安全，高可靠，高可维护，高可用）要求，目前信号系统厂家的传统做法是整个信号系统产品由一家公司来完成，可是随着技算机技术的快速发展，逻辑部份目前已可以采用通用COTS产品，而输入输出部分还是需要各个信号厂家自己设计和生产，因此设计出一款通用型的输入输出控制器已成地铁行业的发展方向。

为了满足以上要求，本文从实际应用角度出发，使信号系统的产品更加的开放透明，设计出基于ARM的地铁用安全型的智能I/O，从而使信号系统设计可以方便地和现场信号设备接口。

在硬件上采用冗余设计，以ARM为主处理器，整个系统无单点硬件故障，采集部分采用动态异或输入设计，驱动部分采用安全驱动设计。

基于ARM的地铁用安全智能I/O严格遵循欧洲铁路信号产品的标准，使系统的安全性，可靠性，可用性和可维护性有了充分的保障。

本文主要介绍了地铁用安全型智能I/O控制器的设计和实现，包括设计思想，具体实施，硬件和软件的设计等。

关键词：冗余，智能，模块化,ARM

Abstract

The input and output equipments of metro signaling system is a interface between logic equipments and field equipments. They require four high requirements (high safety, high reliability, high maintainability ,high availability) Traditionally signaling factory design all signal system, With the development of computer technology, Now logic parts may adopts COTS products, but input and output parts are designed by signaling factory themselves. So it is a development direction for design a general input and output products for metro.

In order to meets above requirements .This discourse is based on real application, The signal products will open and transparent, Design a general intelligent I/O for metro based on ARM, It is convenience interface to field equipments.

We adopts redundancy design in hardware, adopt ARM is main processor, no single hardware fault, Input parts adopts dynamic XOR design, drive parts adopts safety design.

Safety intelligent I/O based on ARM for metro meets European standards for railway, so the products will have high safety, high reliability, high maintainability, high availability.

The discourse mainly introduce safety intelligent I/O based on ARM, include design, realize, design idea, hardware and software and so on.

Keywords: redundancy , intelligent, modularize, ARM

1 绪论

从80年代开始，在运输市场激烈竞争的压力下，各国铁路，特别是发达国家铁路为实现提速、高速和重载运输，积极引进采用新技术，大幅度提高了现代化通信信号设备的装备水平，新型技术系统不断涌现。

1.1 研究背景及其意义

铁路信号控制设备的发展经历了从人工操纵，继电逻辑和计算机控制三个发展阶段，其间经历了一百多年的发展进程，尤其是近年来，随着技算机技术的飞跃发展，在信号设备的控制上出现了许多新技术。同时由于对信号设备的有着最高的安全性，可靠性，可用性和可维护性要求，所以对信号设备的研究和控制有着深远的意义。

1.1.1、故障-安全技术的发展

随着计算机技术、微电子技术和新材料的发展，故障—安全技术得到了飞速发展。高可靠性、高安全性的故障—安全核心设备出现了“双机热备”、“二取二”、“二乘二取二”和“三取二”等不同结构形式，其同步方式有软同步和硬同步。西门子公司、阿尔斯通公司、日本京山公司、日本日信公司等推出了不同类型的采用硬件同步方式的安全型计算机。

故障—安全技术的提高为高可靠和高安全的铁路信号系统的发展打下坚实的基础[1]。

1.1.2、高水平的实时操作系统开发平台

实时操作系统（RTOS,Real Time Operation System）是当今流行的嵌入式系统的软件开发平台。RTOS 最关键的部分是实时多任务内核，它的基本功能包括任务管理、定时器管理、存储器管理、资源管理、事件管理、系统管理、消息管理、队列管理、旗语管理等，这些管理功能是通过内核服务函数形式交给用户调用的，也就是RTOS 的应用程序接口（API,Application Programming Interface)。在过去，由于操作系统的不可控性，在铁路信号设备中严禁使用，随着科学技术的进步，在铁路、航空航天以及核反应堆等安全性要求很高的系统中引入 RTOS，可以有效地解决系统的安全性和嵌入式软件开发标准化的难题。随着嵌入式系统中软件应用程序越来越大，对开发人员、应用程序接口、程序档案的组织管理成为一个大的课题。在这种情况下，如何保证系统的容错性和故障—安全性成为一个亟待解决的难题。基于 RTOS 开发出的程序，具有较高的可移植性，可实现90%以上设备独立，从而有利于系统故

障—安全的实现[2]。另外一些成熟的通用程序可以作为专家库函数产品推向社会，嵌入式软件的函数化、产品化能够促进行业交流以及社会分工专业化，减少重复劳动，提高知识创新的效率。

在铁路这样恶劣工作环境下的计算机系统，对系统安全性、可靠性、可用性的要求更高，必须使用安全计算机，以保证系统能安全、可靠、不间断地工作。而安全计算机系统的软件核心就是 RTOS。目前，英国的西屋公司（Westinghouse）已经在列车运行控制系统中采用了 RTOS，瑞典也有很多铁路通信和控制系统采用 OSE 实时操作系统。

采用实时操作系统可以满足如下性能或特性：

1) 提高系统的安全性。实时操作系统可以成为整个软件系统的中间件，即实时操作系统通过驱动程序与底层硬件相结合，而上层应用程序通过 API 和库函数与实时操作系统相结合。实时操作系统完成系统多任务的调度和中断的执行，这样系统的安全模块和非安全模块将会得到有效的隔离，RTOS 可以很好地解决硬件冗余模块的同步问题[3]。
2) 满足系统实时性的要求。列车运行控制系统要求的是硬实时响应，实时性要求非常高，如果在系统中选用实用操作系统开发该系统的软件，会对该系统的实时性指标的提高有很大帮助[4]。
3) 缩短了新产品的开发周期。由于 RTOS 提供了系统中的多任务调度、管理等功能，在此基础上用户只需开发与应用对象相关的应用程序，所以缩短了新产品的开发周期，降低了设备的成本。RTOS 还具有开发手段可靠、检测手段完善等特点[5]。
4) 充分发挥实时操作系统可移植性、可维护性强等优势。采用 RTOS 后，一旦系统需要升级，只需改动力量程序，而不像以前系统需要重新进行设计，体现出 RTOS 再开发周期短，升级能力强的优点。

1.1.3、数字信号处理新技术的应用

随着铁路运输提速、重载的发展，基于分立元器件和模拟信号处理技术的传统铁路信号设备越来越满足不了铁路运输安全性和实时性的要求。因此，全面引进计算机技术，利用计算机的高速分析计算功能，来提高信号设备的技术水平已非常紧迫。数字信号处理技术（DSP,Digital Signal Processing）的出现为铁路信号信息处理提供了很好的解决方法[6]。

与模拟信号处理技术相比较，数字信号处理技术具有更高的可靠性和实时性。数字信号处理的频域分析和时域分析的两种传统分析方法有着各自的优缺点。频域分析的优点是运算精度高和抗干扰性能好，而缺点是在强干扰中提取信号时容易造成解码

倍频现象，例如将移频的低频11Hz误解成22Hz；时域分析的优点是定型准确，而缺点是定量精确地剔除带内干扰难度大[7]。

随着数字信号处理技术的新发展，在铁路信号处理中引入了新的实用技术，如ZFFT（ZOOM-FFT）、小波信号处理技术、现代谱分析技术等。

目前，我国的轨道电路的信号发送、接收以及机车信号的接收普通采用了数字信号处理技术，日本的数字ATC和法国UM2000数字编码轨道电路也都采用了数字信号处理技术。

1.1.4 计算机网络技术的发展

随着计算机网络技术的飞速发展，实施企业网络化管理已成为企业实现管理现代化的客观要求和必然趋势。

铁路信号系统网络化是铁路运输综合调度指挥的基础。在网络化的基础上实现信息化，从而实现集中、智能管理。

网络化。现代铁路信号系统不是各种信号设备的简单组合，而是功能完善、层次分明的控制系统。系统内部各功能单元之间独立工作，同时又互相联系，交换信息，构成复杂的网络化结构，使指挥者能够全面了解辖区内的各种情况，灵活配置系统资源，保证铁路系统的安全、高效运行。

信息化。以信息化带动铁路产业现代化，是铁路发展的必然趋势。全面、准确获得线路上的信息是高速列车安全运行的保证。因而现代铁路信号系统采用了许多先进的通信技术，如光纤通信、无线通信、卫星通信与定位技术等。

智能化。智能化包括系统的智能化与控制设备的智能化。系统智能化是指上层管理部门根据铁路系统的实际情况，借助先进的计算机技术来合理规划列车的运行，使整个铁路系统达到最优化；控制设备的智能化则是指采用智能化的执行机构，来准确、快速地获得指挥者所需的信息，并根据指令来指挥、控制列车的运行。

近年来，我国铁路行业已成功地推广应用了原TMIS和DMIS（现称TDCS）等系统，在利用信息技术方面取得了长足的进步。具有代表性的列车调度指挥系统TDCS，以现代信息技术为基础，综合运用通信、信号、计算机网络、多媒体技术，建立了新型现代化运输调度指挥系统（铁道部、铁路局、基层信息采集网）[8]。

在欧洲，针对铁路信号网络，专业制定了开放网络传输标准[9]和封闭网络传输标准[10]。

1.1.5 通信技术与控制技术相结合

随着计算机技术（Computer）、通信技术（Communication）和控制技术（Control）的飞跃发展，向传统的以轨道电路作为信息传输媒体的列车运行控制系统提出了新的

挑战。综合利用 3C（Computer、Communication、Control）技术代替轨道电路技术，构成新型列车控制系统已成必然。

用 3C 技术代替轨道电路的核心是通信技术的应用，目前计算机和控制技术已经渗透到列控系统中，称为“基于通信的列车运行控制系统”（CBTC，Communication Based Train Control）。

世界发达国家陆续试验的 CBTC 系统有 ATCS、ARES、ASTREE、CARAT、FZB 等。所有上述各类系统，均具有两个基本特点：

1) 列车与地面之间有各种类型的无线双向通信。可分为连续式和点式的。其中又可分为短距离传输（指 1m 以内）和较长距离传输（远至几公里至几十公里）的移动通信。

2) 它们仍然保留闭塞分区，其中最简易方式 CBTC 仍采用固定的闭塞分区，但是闭塞分区的分隔点不是用轨道电路的机械绝缘节或电气绝缘节（如无绝缘轨道电路），而是用应答器或计轴器，或其他能传送无线信号的装置构成分隔点，这种简易形式仍然保留固定长度的闭塞分区（FAS，Fixed Aotoblock System），简称为 CBTC—MAS。

在 CBTC 中进一步发展的闭塞分区不是固定的，而是移动的（MAS，Moving Autoblock System），简称 CBTC-MAS [11]。

被欧洲联盟采用的 ERTMS/ETCS 的 2 级和 3 级是当前 CBTC 的代表 [12]。

ERTMS/ETCS 经过多个试验项目的测试和认证后，进行了商业项目的建设，德国铁路计划到 2021 年在所有的高速铁路装备 ETCS2 级设备。

通信技术与控制技术的结合重新规划了铁路信号系统的结构与组成，为列车运行控制的未来发展开辟了新开地。

1.1.6 通信信号一体化

随着当代铁路的发展，铁路通信信号技术发生了重大变化，车站、区间和列车控制的一体化，铁路通信信号技术的相互融合，以及行车调度指挥自动化等技术，冲破了功能单一、控制分散、通信信号相对独立的传统技术理念，推动了铁路通信信号技术向数字化、智能化、网络化和一体化的方向发展。

从铁路信号系统纵向发展看，德国已经形成从 LZB、FZB 发展到 ERTMS 的发展趋势。LZB 利用轨道电缆环线传输列车运行控制系统行车指令和速度指令机车信号，取消地面闭塞信号机，保留闭塞分区，列车按固定闭塞方式（即 FAS）运行。FZB 是基于无线的列车运行控制系统，是新一代移动自动闭塞系统（即 MAS），其目的是实现低成本、高性能的列车运行控制系统，并已加入 ETCS。ERTMS/ETCS（欧洲铁路运输管理系统/欧洲列车控制系统）是欧盟支持的统一的行车控制系统，采用 GSM—R 作为

传输系统，其成功应用将进一步推动铁路通信信号的技术进步，加快实现铁路通信信号一体化的进程。从信号系统的横向发展来看，日本新干线在1995年成功开发和投入运行的COSMOS系统，则是通信信号一体化的又一个成功案例。该系统包含运输计划、运行管理、维护工作管理、设备管理、集中信息管理、电力系统控制、车辆管理、站内工作管理等8个子系统，以通信信号一体化技术，实现中心到车站各子系统的信息共享，并使系统达到很高的自动化水平。另外成功地应用了安全光纤局域网，使之成为联锁系统、列车运行控制系统的安全传输通道，达到通信技术与信号安全技术的深度结合，实现了通信信号一体化[13]。

通信信号一体化是现代铁路信号的重要发展趋势，铁路信号技术发展所依托的新技术，如网络技术，与通信技术的技术标准是一致的，属于技术发展前沿科学，为通信信号一体化提供了理论和技术基础。在借鉴世界各国经验的基础上，结合中国国情、路情，我国已制定了中国统一的CTCS技术标准（暂行）。

1.1.7 安全性与可靠性分析

保证铁路运输的安全，要求铁路信号系统具有高可靠性和高安全性。安全评估理论的建立与推广为定量评估铁路信号系统的可靠性和安全性提供了重要手段[14]。

在故障—安全理论的发展上，20世纪90年代初，IEC（International Electrician Committee，国际电工委员会）将故障—安全的概念进行了量化，制定了安全相关系统的设计和评估标准IEC61508。该标准提出了安全相关系统的“安全完善度等级（SIL,Safety Integrity Level）”的概念，它是一个对系统安全的综合评估指标。

IEC61508对安全系统提出了如下要求：

1) 功能性（Functionality），包括容量和响应时间；
2) 可靠性和可维护性（Reliability and Maintainability）；
3) 安全（Safety），包括安全功能和它们相关的硬件/软件安全完善度等级（SIL）；
4) 效率性（Efficiency）；
5) 可用性（Usability）；
6) 轻便性（Portability）。

随后欧洲和日本相应地以IEC61508标准为基础，制定了相关的信号系统的设计评估标准以及安全认证体系。

欧洲电工标准委员会（CENELEC）基于IEC61508标准为基础，附加列车安全控制系统的技术条件制定了一些安全相关系统开发和评估的参考标准。这些标准包括：

1) EN50126铁路应用：可信性、可靠性、可用性、可维护性和安全性（以下简

称 RAMS）规范和说明；

2） EN50129 铁路应用：信号领域的安全相关电子系统；

3） EN50128 铁路应用：铁路控制和防护系统的软件。

欧洲是世界上铁路最发达的地区之一。欧洲国家多，国土面积小，各国内部的铁路网很密集。近几年来，欧洲铁路公司和信号公司在对各自的既有信号系统进行升级或者技术改造的同时，在欧盟(EU)委员会和国际铁路联盟(UIC)的推动下，欧洲 7 大铁路信号公司，如法国的 Alstom(阿尔斯通)公司、瑞典的 Adtranz 公司、德国的 Siemens(西门子)公司、法国的 Alcatel(阿尔卡特)公司、意大利的 Ansaldo(安萨尔多)公司(含法国 CSEE 公司)、英国 WestingHouse(西屋)公司，以及 Invensys 公司，联合起来为信号系统的互联和兼容问题制定信号标准，并制造了相关的产品。

在进路控制方面，随着区域计算机联锁技术逐步取代陈旧技术，自动化系统得到广泛应用。

在列车防护和控制系统方面，研制了基于通信的列车控制系统(CBTC)。

铁路信号计算机联锁系统是一种以计算机为主要手段实现车站联锁的系统，是保障行车安全的基础设备之一。随着我国铁路不断提速和运输效率的提高，以及地铁在我国各大中城市的迅猛发展，对计算机联锁的安全性及时间响应的要求不断提高[15]。

1.2 研究的目的

随着铁路运输朝着高密、重载及高速的方向发展，既有的车站铁路信号联锁装置已无法适应铁路信号对可靠性与故障——安全性的更高要求。就技术方面而言，铁路信号系统已经历了机械联锁、电气联锁（继电联锁）等二个阶段，目前在我国干线铁路或企业自备铁路上所使用的联锁系统绝大多数仍为继电联锁系统。70 年代末期新型微处理器的出现以及容错理论与技术的逐步完善，激励人们以微型计算机为核心构成计算机联锁系统[16]。

目前，我国国铁和地铁计算机联锁一般采用双机热备、三取二、二取二乘二的冗余结构。系统一般由上位机、联锁机、I/O接口和维修机以及其他一些附属设备组成。

上位机主要供行车调度人员使用，发送操作命令和显示现场设备的状态。

维修机主要是记录各种操作命令和设备状态，给设备维护人员提供维护帮助。维修机作为上位机的备用机，在上位机发生故障时，经授权、登记，维修机替代上位机工作。

联锁机是计算机联锁平台的核心，它从人机会话层接受命令来完成联锁功能，主要进行联锁逻辑及运算判断，输出相关的安全的控制命令和有关设备状态的表示信息，输入输出 I/O 是联锁机的重要组成部份。

研制地铁用智能 I/O 基于以下目的：

1） 研究通用型智能 I/O 也是计算机联锁实时性要求，联锁系统必须不失时机地

采集到输入变量的变化情况，及时刷新站场各类表示信息，及时输出道岔和信号的控制命令，而且对涉及安全的控制命令必须以具有故障——安全特征的形式输出[17]。

2) 研究通用型智能 I/O 也是联锁设备需要高的可靠性与故障——安全性的要求，信号联锁系统是一种实时控制系统，它必须是高可靠的，通常继电联锁系统在采取预防性维护措施的前提下其 MTBF 可达 1.3×105h[2]（约 15 年），采用工业级的控制计算机与容错技术完全可以达到并超出这一指标[18]。

3) 设计通用型智能 I/O 也是系统内信息传递的可靠性与安全性的要求，鉴于工业计算机自身不具备故障——安全特性，因此系统内传递的信息也不具备安全性，受各种干扰、辐射以及各类故障的影响，信息畸变在所难免，从而造成逻辑运算错误而可能引发危险侧输出[19]。

4) 系统内信息变换及逻辑运算的安全性：就联锁程序而言，无论设计调试方法多么严密也很难排除所有隐含的缺陷[20]，这就要求必须引入避错及容错机制使故障形成的危险侧运算结果输出的概率达到规定的要求，因而需求要对通用型智能 I/O 加以研究。

5) 研究和设计地铁专用的通用型智能 I/O 也是经济性的要求，计算机联锁系统取代继电联锁系统的另外一个重要原因是为了降低系统费用成本，一般来说系统费用表现在设计、制作、施工、调试以及建筑费用上，因此计算机联锁系统必须在以上若干方面充分显示其优势。

6) 研究和开发通用型智能 I/O 也是功能扩展的要求，旧有的继电联锁系统只能提供基本联锁功能与操作界面，新型计算机联锁系统除此之外，还应具有故障诊断与分析、重演、远程通信及其他管理功能[21]。

计算机联锁系统的I/O接口部分是保证联锁系统的安全性和可靠性的重要组成部分，使I/O部分实现智能化不但可以提高联锁系统的安全性和实时性，同时使I/O部分实现通用化，应用到其它信号产品的I/O部分，如车载设备等。因而具有广阔的应用价值。

I/O 接口是联锁系统与控制室外信号设备的继电器之间的接口，是涉及到安全的部件之一。使 I/O 部分实现智能化对提高整个联锁系统的工作效率和安全性都有重要的意义。

1.3 论文的结构

本论文分为五个部分，第一部分为绪论，介绍了研究的背景资料，阐述了基于 ARM 的 I/O 控制器研究的背景、目的和意义；第二部分介绍了安全型智能 I/O 的处理理论；第三部份介绍了实施方案，第四部份介绍了 I/O 控制器的硬件结构；第五部份介绍了模块的软件结构；最后为研制总结。

2 安全型智能 I/O 处理理论

2.1 计算机联锁系统的基本结构

由于计算机联锁系统的综合性能远远超过继电联锁系统，因此车站联锁系统由继电装置向计算机联锁系统转化已成为一种不可扭转的趋势。具体来说计算机联锁系统的优势主要表现在适时性、安全性、可靠性、可维护性及性价比等若干方面 [22]。

计算机联锁系统是一套专用的硬件与软件系统，它是实现信号、进路与道岔间的联锁关系的安全计算机设备，计算机联锁系统实质上是一个满足故障——安全信号原则的联锁逻辑运算系统，计算机在系统中的作用是将操作命令与现场各种输入的表示信息读入，再根据计算机内部状态等条件进行逻辑运算，判断后输出控制信息至执行机构。目前，我国铁路车站计算机联锁系统多采用双机热备、三取二、二取二乘二的冗余结构。其中双机热备方式，平时一套系统工作，另一套系统备份，当正常工作的系统出现故障时，切换到备份系统工作；三取二的系统则采用三套系统同时工作，输入输出则采用三取二的原则，即三个结果中只要有二个正确，则系统进入下一个流程；二乘二取二则常是采用两套系统互为热备，每套系统中使用二套硬件，只有两套硬件的计算机结果完全正确，则进入下一步计算。三种方式在计算机联锁系统中都有成熟应用，也各有优缺点 [23]。

2.2 地铁用安全型智能 I/O 处理概述

铁路信号计算机联锁系统是一种以计算机为主要技术手段实现车站联锁的系统，是保障行车安全的重要设备之一。随着我国铁路不断地提速和运输效率的提高，对计算机联锁的安全性及响应时间的要求也越来越高。计算机联锁系统的 I/O 接口部分是保证联锁系统的安全性和可靠性的重要组成部分，使计算机联锁逻辑使用通用 COTS 产品，使 I/O 部分实现智能化和通用化是信号产品的发展趋势，这样不但可以提高整个联锁系统的安全性和实时性，还可以扩展联锁系统的功能，同时使 I/O 接口部分为地铁通用的输入输出控制产品。

2.3 智能 I/O 在信号系统中介绍

智能 I/O 在信号产品中所处的地位。

一般的计算机联锁系统是由上位机、联锁机、I/O 接口和维修机以及其他一些附属设备组成，其系统结构功能见图 2.3.1。

其中 I/O 部分是联锁控制逻辑设备与继电控制设备之间的接口，在图中用绿色部份表示，带有阴影表示采用了硬件冗余结构，它从联锁控制逻辑部份接收控制命令，将外部设备的状态信息传给联锁控制逻辑。输入输出控制器从继电器接口采样输入，

输出驱动继电器。

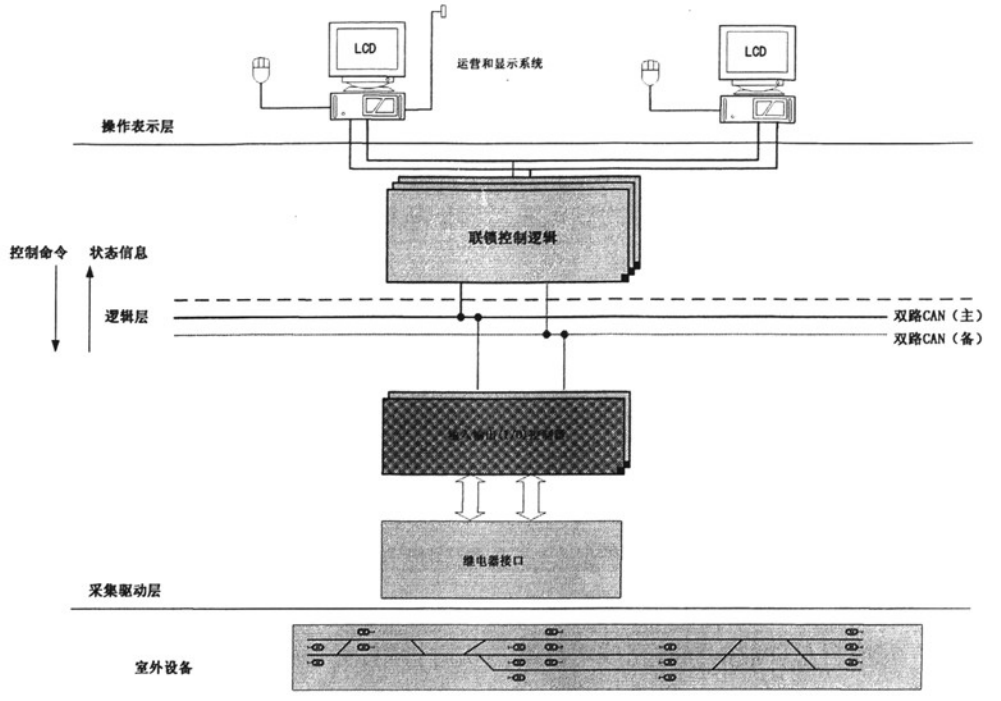

图2.3.1　计算机联锁系统结构

图 2.3.1 中的阴影部分表示采取了冗余结构，如联锁控制逻辑部份采用三取二，输入输出控制器采用热备的方式。

2.4　问题的提出

目前，为满足联锁系统的安全性和可靠性的要求，国内联锁系统的 I/O 接口部分大多采用动态采集和驱动动态继电器的方式来完成。系统中采用了动态循检的方式保证 I/O 部分的安全性和可靠性，但这种 I/O 接口仍存在以下不足：[24]

1) I/O 部分的功能实现是由联锁机来保证，占用了大量的联锁机的 CPU 资源，增加了周期运行时间；

2) 动态的 I/O 响应慢，一致性差；

3) 动态的 I/O 对于脉冲群的干扰和间歇性的故障防护困难；

4) 自检速度慢，故障检测覆盖率低；

5) 动态 I/O 的单套硬件本身不完全是故障－安全，尤其是采集/ 驱动与联锁 CPU 总线的接口板，并不具备故障－安全特性。

6) 不同联锁系统的I/O部分也各不相同，难以达到通用化，不利于系统改造和新产品开发。

在国外一些采用区域联锁控制的车站，I/O 部分已经实现了智能控制，并且演变为现场设备控制部分，所以可以看出，使I/O部分实现智能化是完善联锁功能，提高系统的安全性和实时性的一种行之有效的解决方案。同时研制和开发实用于国内联锁系统的安全智能型I/O模块也为联锁系统的更新和升级，以及开发区域联锁系统打下了良好的基础。

2.5 I/O的设计思想

近几年，高稳定性、高可靠性的处理器广泛应用于航空、铁路以及工业控制领域。由于 I/O 模块的功能相对于联锁机来说比较简单，所以其 CPU 部分选用有高可靠性的处理器最合适。联锁机与 I/O 模块的信息传输选用了 CAN 现场总线，因为这种总线结构在铁路领域已经被广泛应用于信息传输方面，由于其极高的可靠性和独特的设计，以及高速率、传输距离较长的特点，特别适合工业现场监控设备的互连 [25]。

I/O模块采用了智能化控制动态采集和静态直流驱动，同时加入了动态自检，使模块在完成自身功能的同时完成对故障的实时检测。另外为了消除驱动模块中驱动电源对模块和联锁系统安全性的影响，使其符合故障-安全原则，在充分考虑了各种故障可能的情况下，提出了由CPU联合控制的鉴相安全电源的解决方案，提高了模块和联锁系统的安全性。

3 实施方案

基于ARM的智能I/O采用故障——安全模式设计。一个模块由两个功能相同的单元构成，两个单元的输出数据一致时，才向外输出数据。如果输入数据比较不一致，则不对数据进行处理。当模块发生故障时，采取安全处理措施，切换到备份模块。对于开关量数据，由I/O板中的两个子模块用两个输出点，分别控制安全继电器的两条控制线，实现安全输出。

3.1 体系结构

常见的安全计算机系统结构冗余方式，如双机热备、三取二表决、二取二表决、二取二乘二等，它们的安全性和可靠性指标已有很多文献从理论上进行了论证，并且已经应用在多种轨道交通信号设备上，如车站计算机联锁系统和列车自动运行系统等。目前，采用双机热备形式的安全计算机应用比较广泛，在双机热备的技术基础可以构建二取二乘二的结构，而两者相比，后者明显具有更高的安全性和可靠性[26][27]。

二取二乘二系统分为相互独立的A、B两个运算系，每系内均有二个关系对等的主处理单元，每个主处理单元均能同时接收接口层的输入信息。在信息处理与计算过程中，同一运算系内的二个主处理单元通过诊断机制进行同步校验，所有信息校验无误后软件方可输出动态码型的控制信号给本系内的二取二输出表决模块，表决一致才能够输出有效的控制信号。两个运算系分别独立地完成相同的任务，而由专门的主备切换单元判断两系的运行状态是否正常，执行主备切换操作，并将主运算系的控制信号输出给被控设备。

3.2 系统RAMS指标

由于欧洲铁路有着一百多年的历史，对铁路信号产品也有着一套完整的标准，目前我国城市轨道交通产品的业主已明确要求信号系统的产品符合欧标，因此本产品的开发严格按照欧洲相关信号标准执行。

在欧洲安全标准中，对信号产品的RAMS指标作出了严格的规定：[28]

1) 安全性要求：安全性指标用故障情况下倒向危险侧的概率来表示

表3.2.1 信号子系统安全完善度等级要求

ATC子系统	故障导向危险侧的概率	安全完整性水平
ATP	10^{-9}	4
ATO	10^{-7}	2
联锁（含计轴）	10^{-9}	4
ATS	10^{-7}	2

表3.2.1中列出了信号系统中各个子系统对安全性的要求，如联锁子系统，故障

导向危险侧的概率为 10^{-9}，安全完整性水平要求 4 级，也就是说出了 10^{9}个故障，只允许有一个故障是非安全的，因为绝对的安全系统是不存在的，只有把安全性指标量化，才有助于我们对安全产品的研究和控制。

2) 可靠性：联锁子系统的单机系统或通信通道：MTBF≥2.0×10^{5} h；

计算机外围设备的 MTBF≥5×10^{4} h；

3) 可用性：各子系统和单项设备的可用度应≥99.9998%；

整个信号系统的可用度应≥99.999 %。（每 2.85 年故障中断运营时间累计不超过 15 分钟）

4) 可维护性（故障修复时间）：车载设备、控制中心设备、车站设备、轨旁设备（转辙机除外）的 MTTR≤15 分钟；

影响铁路 RAMS 的因素 [29]

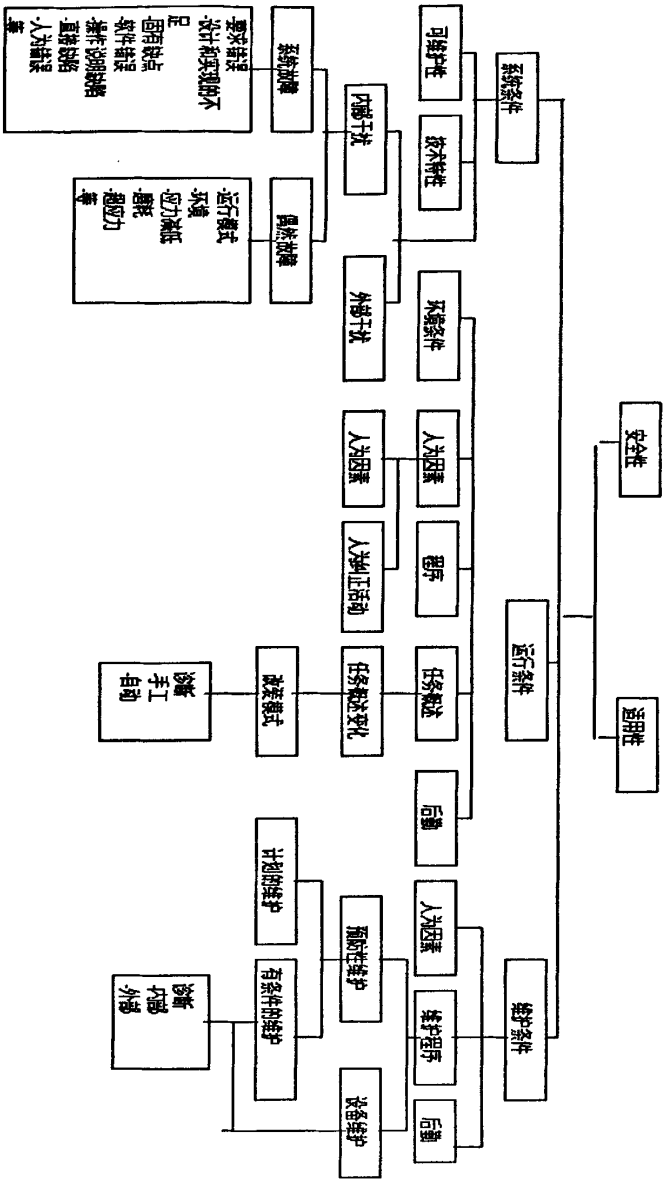

图 3.2.1　影响铁路 RAMS 因素

图 3.2.1 表明了影响 RAMS 的因素主要从安全性和适用性两个方面，受影响的条件分为系统条件，运行条件和维护条件。如对于系统条件来讲，系统可能受到内部和外部干扰，对于内部干扰来说，可能会导致系统故障和偶然故障，对于系统故障来说，原因可能是要求错误，设计和实现的不足等原因造成的。

故障的分类如表 3.2.2：

表 3.2.2 RAM 故障分类

故障分类	
重大的（固定的故障）	故障：造成列车无法移动或造成服务延误超过规定时间和／或产生超过规定水平的费用
主要的（运行故障）	故障：为使系统达到规定的性能必须做出调整和造成的延误或费用超过重大故障规定的最低限度
较小故障	故障：没妨碍系统达到其特定的性能和不符合重大或主要故障的标准

表 3.2.2 列出的故障的分类，可分为重在的，主要的和较小的故障，并对每类故障进行了严格的划分。

3.3 I/O 控制器机架结构设计

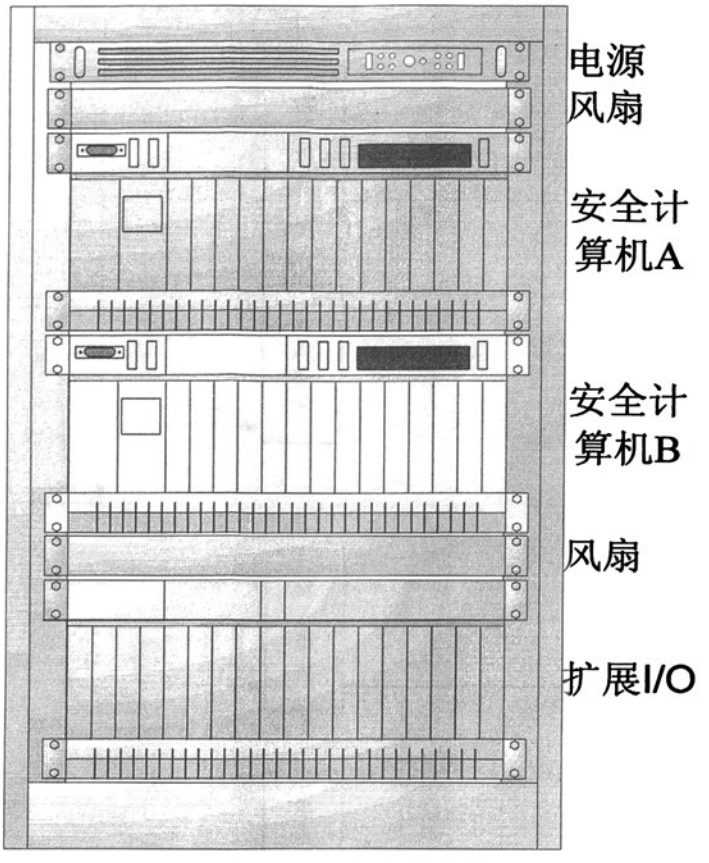

图 3.3.1 机架结构图

控制器机架结构如图 3.3.1，最上一层枢机框为 I/O 控制器的一个系，第二层机框为第一层的热备，整个结构为 2×2 取 2 结构，最下一层为继电器单元，主要用于系统的调试，它用来模拟继电器柜里的所有继电器，机架和机箱为 ELMA 公司的产品，一个机架内安装三个机箱，机箱内有内置导轨，CPU 模块和 I/O 模块从前面插入通过导轨插入，前面板上有电源开关和状态指示灯，后面出线，机箱不盖后面板。

最上层第一个机框内第一个模块为电源模块，第二、第三个模块为 CPU 模块，组合成二取二架构，第四个模块为 COM 模块，负责对所有模块的监视和对外部设备的通讯。

I/O 控制器部分实现原理：两台 2 取 2 互为热备的控制器对接收来的数据进行处理，再通过 CAN 总线传送到 I/O 板，完成驱动外部继电器，输入外部继电器接点状态。

这两个控制器相互监视，分为主和备控制系统；两个控制器的切换可通过指令完成，也可通过控制器本身完成。

3.4 外部输入的连接

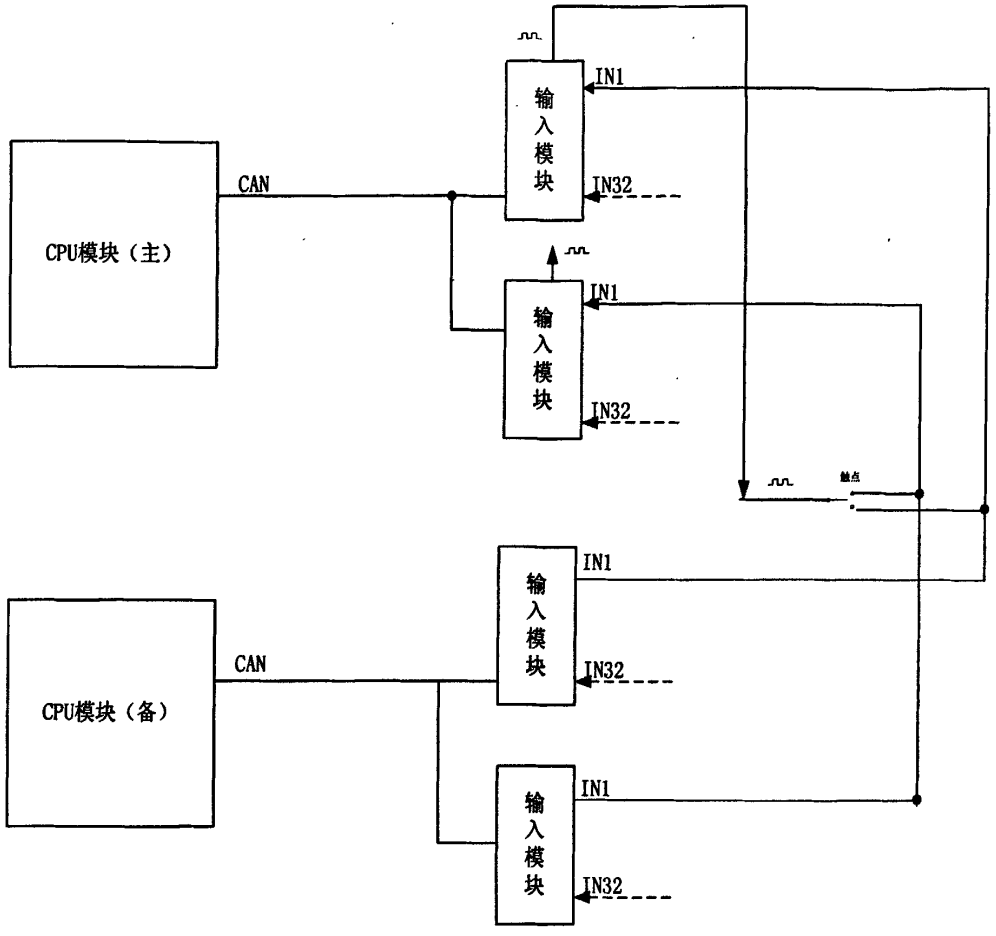

图 3.5.1 外部输入连接图

如图 3.5.1 所示，节点采集通过 CPU 模块向公共端输入动态方波信号，输出信号经节点的常开或常闭节点到输入模块，输入模块由两个相同 I/O 模块组成，每个模块把采样输入信号通过 CAN 总线传送给 CPU 模块，由输入模块完成对输入信号的二取二控制。

3.5 外部输出的连接

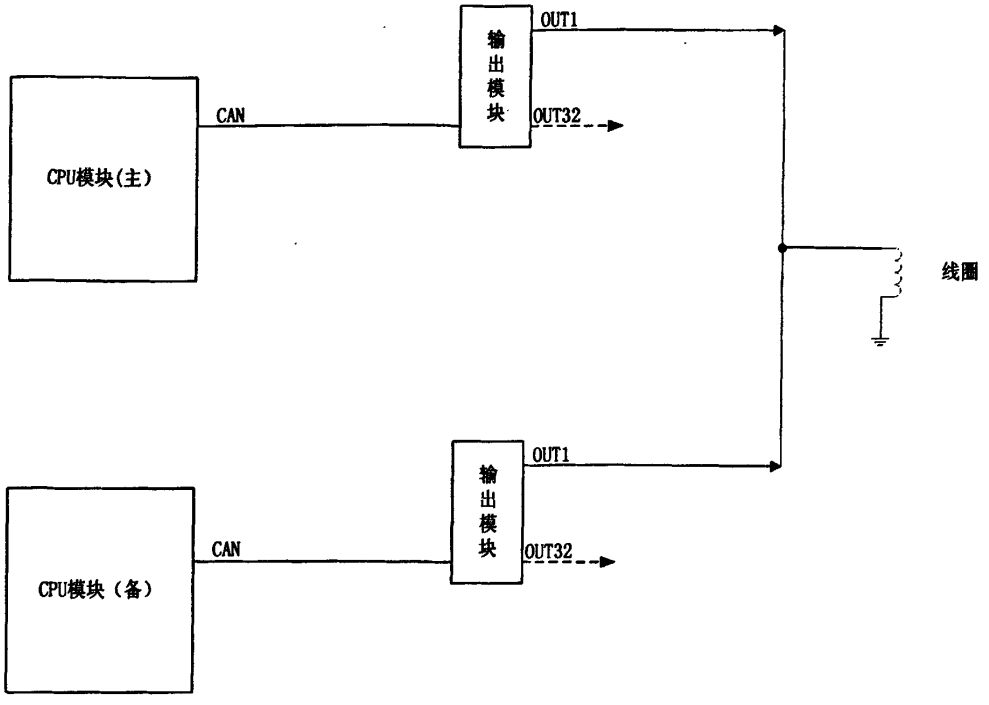

图 3.6.1 外部输出连接图

图 3.6.1 为外部输出的连接方式，当采用继电器的线圈一端接地的驱动方式时，由主模块完成二取二的输出，备 CPU 模块虽然也连接到负载上，但并不驱动，只有当主系出现故障时，备系切换成主系时，才产生有效输出。

3.6 主要完成以下功能：

1) 2 乘 2 取 2 的冗余管理；
2) 热备主从的切换（自动或手动）；
3) 主从状态指示；
4) 故障时对 CPU 模块电源的关断；
5) 和三取二主机通讯；
6) 对 I/O 模块控制；
7) 输入、输出点的冗余配置；
8) 对 CAN 总线的管理；
9) I/O 控制器的自检；

3.7 环境及可靠性设计

3.7.1 模块化设计

为了获得高可靠性、良好的维修性和可生产性，采用了模块化系统设计。

3.7.2 热设计

热设计是一个重要的技术问题。该问题的解决主要基于下列技术措施：

1) 采用低功耗 CMOS 器件；

2) 采用导热约束阻尼印制板设计技术。印制板和器件之间设有金属导热层，以便把器件的功耗热量有效地传导至机箱壳体并向外散发，保证机内温度比环境温度不高于 15℃；

3) 发热大的器件上表面外加散热片

3.7.3 抗振设计

设计两级减振机制，即整机减振和模块板减振。整机设计外减振器，模板级则采用阻尼印制板，在印制板与导热层之间又设置了阻尼层，印制板采用加强条等力学加固措施。这些设计技术的采用，保证了整机良好的力学环境适应性。

3.7.4 隔离设计

采用抗干扰隔离设计是提高故障检测处理器工作可靠性的关键技术之一。设计中将按照任务书的要求实现输入接口、输出接口与主机部分的光耦隔离。在通信部分也采取隔离措施。

3.7.5 冗余设计

为满足可靠性指标要求采用冗余设计，采用三取二表决式三机容错体系结构和 2 乘 2 取 2 冗余体系结构，消除单点。硬件同步的方案可以满足系统实时性要求。在实际设计中要充分借鉴同类产品的设计方法。

3.7.6 降额设计

采用降额设计，使元器件的工作条件都留有充分的余地，提高产品的可靠性。降额参数包括电压、电流、功耗、频率、负载能力等电气参数，压力、振动、冲击等机械参数和温度、湿度等环境参数。

3.7.7 潜通路防护设计

潜通路主要是在开关量冗余输入接口及单机的互连线中产生，为保证冗余互连的有效性，做好潜通路的分析与防护工作，防止某单机电源故障时影响相关信号。在设

计前期，认真分析可能发生潜通路的地方，设计有效的防护电路，并选用防止潜通路的接口器件。

3.7.8 安全性设计

由于安全计算机为低压工作设备，在生产、调试与试验中不会危及到人员生命，不会对环境造成污染，也不会影响其他设备的正常工作，不存在相应的危险源。产品安全性分析主要是在可靠性分析工作的基础上，按照甲方安全性分析要求，将针对产品任务阶段重点进行故障模式、影响分析（FMEA）；并在分析的基础上对确定系统是否存在Ⅰ、Ⅱ类单点故障，将完成《成败型、灾难性故障模式及对策分析报告》。

3.7.9 电磁兼容设计

1) 在使用TTL电路时，在每片组件和印制板电源输入端都接上了去耦滤波电容。
2) 采用了高抗干扰直流稳压电源。
3) 整机采用单点接地，模拟地和数字地分开，二地在电源处合接。
4) 在元件布局和布线时，注意输入线和输出线、系统电源和直流电源线不平行、不靠近，各元器件之间连线尽可能短。不同的连接线分开。
5) 增加了电源滤波器。

3.7.10 电子器件管理

1) 关重件有交货质量检验。
2) 所有电子元件百分之百进行老化筛选。

3.7.11 系统的热设计

1) 机箱的设计充分考虑机器的散热问题。
2) 内部采用风扇散热的方法。在设计时，还考虑了元器件本身的散热问题，并采取了一定的措施，使之容易自然散热。
3) 选用温度特性好的元器件，集成芯片选用工业级，电阻和电容等其他元器件温度特性也都有足够的余量。

3.7.12 维修性设计

为了便于维修，所有板件和接插件都易于检查和替换，整机具有很好的维修可达

性，更换备件板即可正常工作。

1) 电源采用模块化设计，更换维修简单易行。

2) 插件、接口、插头座、电缆均有编号，标记清晰醒目，不易失色和脱落。

3.8 环境条件要求

1) 工作环境温度：-25℃- +60℃

2) 贮存温度：-45℃- +85℃

3) 环境湿度：95%±3%（+35℃时）

4) 可以承受公路、铁路、空运和船运的运输环境。

4 硬件结构

安全智能型I/O模块的硬件主要包括CPU部分、CAN总线通信部分、采集或驱动电路部分、报警和信息显示以及电源部分，系统的硬件采用二取二结构,也可根据业主需要组合成三取架构。

在这样的多重冗余结构中，各单元间的通信接口需求较为复杂，通信任务较为繁重。同时，软件除了要完成正常冗余处理任务功能，还要进行容错与避错处理。因此，选用ARM 处理器为核心，设计主处理单元以实现上述结构。CPU主处理器采用PHILPS公司ARM7系列产品LPC2294。

LPC2292/LPC2294 微控制器是基于一个支持实时仿真和嵌入式跟踪的16/32位ARM7TDMI-S CPU，并带有256KB嵌入的高速Flash存储器。128位宽度的存储器接口和独特的加速结构使32 位代码能够在最大时钟速率下运行。对代码规模有严格控制的应用可使用16 位Thumb 模式将代码规模降低超过30%，而性能的损失却很小。

由于LPC2292/LPC2294的144 脚封装、极低的功耗、多个32位定时器、8通道10 位ADC、2/4（LPC2292/LPC2294）高级CAN、PWM通道以及多达9个外部中断管脚使它们特别适用于汽车、工业控制应用以及医疗系统和容错维护总线。LPC2292/LPC2294 含有76（使用了外部存储器）到112个（单片）可用GPIO口。由于内置了宽范围的串行通信接口，它们也非常适合于通信网关、协议转换器以及许多其它的通用应用中[30]。

ARM2294的主要特点:

1) 16/32位ARM7TDMI-S 微控制器，LQFP144封装。
2) 16 kB 片内静态RAM和256kB片内Flash 程序存储器。128位宽接口/加速器可实
3) 可高达60MHz的工作频率。
4) 通过片内引导装载程序软件实现在系统编程/在应用编程（ISP/IAP）。
5) 区或整片擦除时间为400ms以及256字节的编程时间为1ms。
6) 嵌入式ICE-RT和嵌入式跟踪接口使用片内RealMonitor 软件对任务进行实时调试。
7) 并且支持对执行代码进行高速实时跟踪。

8) 2/4(LPC2292/LPC2294)个互连的 CAN 接口，带有先进的验收滤波器。多个串行接口，包括 2 个 UART(16C550)、高速 I2C 接口（400 kbit/s）和 2 个 SPI 接口。

9) 8 路 10 位 A/D 转换器，转换时间低至 2.44μs。

10) 2 个 32 位定时器（带 4 路捕获和 4 路比较通道）、PWM 单元(6 路输出)、实时时钟(RTC)和看门狗。

11) 向量中断控制器(VIC)。可配置优先级和向量地址。

12) 通过外部存储器接口可将存储器配置成 4 组，每组的容量高达 16MB，数据宽度为 8/16/32 位。

13) 多达 112 个通用 I/O 口（可承受 5V 电压）。多达 9 个边沿或电平触发的外部中断引脚。

14) 通过片内 PLL 可实现最大为 60MHz 的 CPU 操作频率，设置时间为 100us。

15) 片内晶振频率范围：1MHz～30 MHz。

16) 2 个低功耗模式：空闲和掉电。

17) 通过外部中断将处理器从掉电模式中唤醒。

18) 可通过个别使能/禁止外部功能来优化功耗。

19) 双电源,CPU 操作电压范围：1.65V～1.95 V(1.8 V± 0.15 V),I/O 操作电压范围：3.0V～3.6 V(3.3 V± 10%)，可承受 5V 电压。

ARM7TDMI-S 是一个通用的 32 位微处理器，它可提供高性能和低功耗。ARM 结构是基于精简指令集计算机(RISC)原理而设计的。指令集和相关的译码机制比微编程的 CISC 要简单得多。这样使用一个小的、廉价的处理器核便可实现很高的指令吞吐量和实时的中断响应。由于使用了流水线技术，处理和存储系统的所有部分都可连续工作。通常在执行一条指令的同时对下一条指令进行译码，并将第三条指令从存储器中取出[31]。

ARM7TDMI-S 处理器使用了一个被称为 THUMB 的独特的结构化策略，它非常适用于那些对存储器有限制或者需要较高代码密度的大批量产品的应用。在 THUMB 后面一个关键的概念是“超精简指令集”。ARM7TDMI-S 处理器基本上具有两个指令集[32]：

1) 标准 32 位 ARM 指令集

2) 16 位 THUMB 指令集

THUMB 指令集的16 位指令长度使其可以达到标准ARM 代码两倍的密度，却仍然保持ARM 的大多数性能上的优势，这些优势是使用16 位寄存器的16 位处理器所不具有的。这是因为THUMB 代码和ARM 代码一样，在相同的32 位寄存器上进行操作。

THUMB 代码仅为ARM 代码规模的65%，但其性能却相当于连接到16 位存储器系统的相同ARM 处理器性能的160%。

LPC2294 分别包含4 个CAN 控制器。CAN 是一个串行通信协议，它能有效支持高安全等级的分布实时控制。CAN 的应用范围很广，从高速的网络到低价位的多路接线都可以使用CAN。

ARM2294 上CAN 的特点：

1) 单个总线上的数据传输速率高达1Mb/s

2) 32 位寄存器和RAM 访问

3) 兼容CAN 2.0B, ISO 11898-1 规范

4) 全局验收滤波器可以识别所有CAN 总线的11 位和29 位Rx 标识符

5) 验收滤波器为选择的标准标识符提供了FullCAN-style 自动接收

利用LPC2294 上的CAN 总线接口逻辑进一步提高系统的可靠性和安全性，主要的外部数据通信接口，包括CAN 总线、RS-232 总线、RS-422 总线以及10/100M 以太网均为双套冗余设计。同时，系统还提供了看门狗定时器（WDT）、工作电压监测复位和实时时钟（RTC）等功能。安全计算机位于信号设备的安全处理与运算层，它除了双重冗余的10/100BaseT 以太网用于同上位机进行通信外，系统中每个主处理单元均各自设计了2 路RS-232 接口、2 路RS-422 接口、2 路CAN2.0 接口，它们预留作为与其他轨道交通信号系统的通信接口[33]。

安全I/O 模块专门用于控制和采集轨道交通信号设备的开关量信号，它具有动态信息码处理能力，电路设计符合故障—安全特性。它的输入和输出通过表决后即可控制设备接口层的现场设备。处理器通过由CAN 总线扩展安全 I/O 模块，这种总线对所输出的数据和地址具有回读校验能力。

地铁用安全型I/O 由主机框部分和扩展机框部分组成。当主机框安装的I/O 接口板数量不能满足需求时，使用若干个扩展机框用于安装I/O 接口板。I/O 板每两块一组，构成热备冗余通道。安全计算机整机安装于机柜的若干个机框中。最大可扩展故障——安全输入/输出开关量设计容量1024 路，故实际使用时，可根据实际需要，配

备不同数量的扩展机框。

硬件系统采开放灵活的设计，可以根据业主和工程的实际需要，把硬件系统配置成3取2或2取2乘2的冗余方式。

当系统配置为2乘2取2模式时，安全计算机主机部分（不包括I/O接口板扩展机框）为两个机框，每个机框内安装2块CPU板、1块通讯板和若干块I/O接口板。

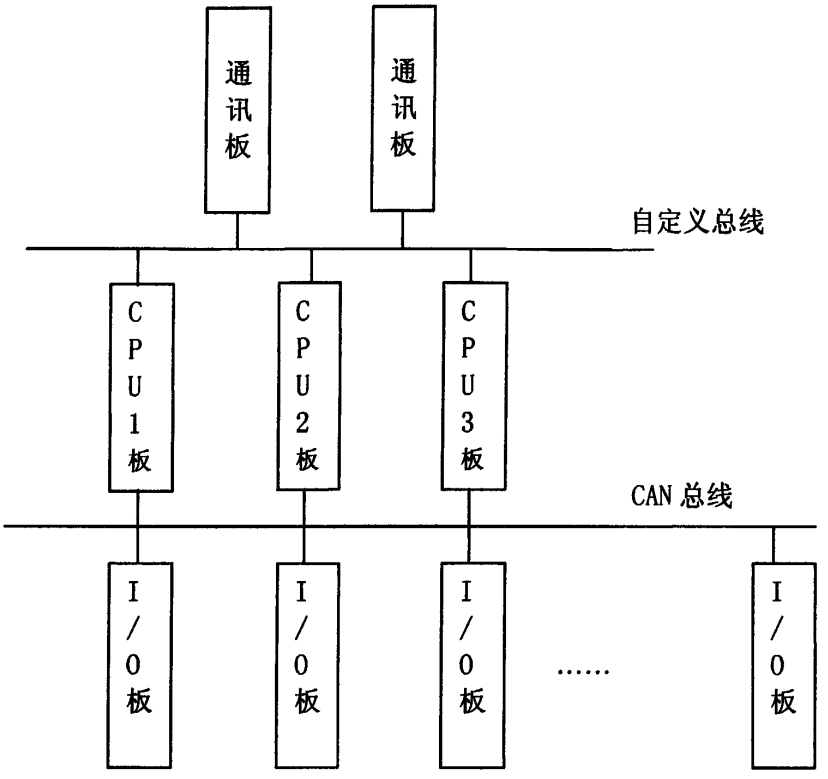

图4.1 3取2冗余安全计算机结构框图

当系统配置为3取2模式时，安全计算机主机部分为1个机框，机框内安装3块CPU板、2块通讯板和若干块I/O接口板。3取2冗余安全计算机结构如图4.1所示。

3路+24V、3路+5V以及光耦接口部分电源由外部模块提供，本设计不予考虑。

3块CPU板结构完全相同，CPU板之间通过同步串口交换数据，CPU板与通讯板之间通过FIFO交换数据。通过同步电路实现3块CPU板的操作同步。3块CPU板互相传递程序运行信号、互相监视，切断发生故障的CPU板的电源，以便维修。

CPU板与I/O接口板之间采用CAN总线通讯，每块I/O板有4路CAN总线接口：当系统配置成3取2冗余结构时，使用其中的3路；当配置为2乘2取2模式时，其

中2路工作，另外两路作为备用。

支持3取2冗余模式的措施：

1) CPU板之间通过同步串口交换数据；
2) CPU板与通讯板之间通过FIFO交换数据；
3) 通过同步电路实现操作同步。

支持2乘2取2模式的措施：

1) 2取2模式：一个模块由两个功能相同的单元构成，两个单元的输出数据一致时，才向下一级模块或向系统外部外输出数据。
2) 2乘2取2模式相当于在2取2冗余设计的基础上增加热备份。主、备之间相互监视对方工作状态，可相互切换、通过上一层模块切换或手动切换。主、备处理器都工作。主板输出数据；备板不输出数据。

支持热插拔的措施：

1) 借鉴支持热插拔的CPCI接插件的电气规范，对于电源、一般信号、使能信号，分别采用长、中和短针，保证板卡在供电正常的情况下，才开始工作；
2) 在3块CPU交换数据的接口处，采用LSTTL器件。保证在插拔瞬间，在接口两端电源电压存在较大差别的情况下，不存在潜通路；
3) 控制信号入口处采用反串二极管和上拉电阻的方式，防止潜通路，支持热插拔；
4) CPU板与I/O接口板之间采用CAN总线通讯，CAN总线控制器采用PHILIPS公司的SJA1000，其PeliCan模式支持CAN2.0B协议规定的所有功能，支持热插拔（Hot Plugging Support）[34]；
5) 同时软件要有监视各板卡存在与否状态的功能。

4.1 CPU模块

I/O控制器的CPU采用PHILIPS公司的ARM7系列，并且在模块中采用双CPU结构。

在本项目CPU采用LPC2294FBD ARM微控制器，

I/O控制器CPU板特性：

1) 在本系统中采用32位ARM7TDMI-S核，应用程序编程语言为ARM32位汇编加C语言组成，主要为C语言；
2) 16 kB片内SRAM数据存储器,256kFLASH程序存储器；
3) 通过UART0装载片内boot程序提供在线系统下载及编程；

4) 使用片内 RealMonitor 软件对前台任务进行调试时，中断服务程序可继续运行；
5) 使用 1 个 UART(16C550)完成双机热备，用 1 个 RS422 和上位机三取二通讯；
6) 使用 10 位 A/D 转换器完成对电源电压的采集控制，转换时间低至 2.44μs；
7) 采用 ARM7 系列 LPC2294 作为 CPU 芯片，作为 I/O 控制器的主处理模块，工作频率 24MHz。

CPU 主要负责整个模块的控制以及与联锁机之间的通信。在采集模块中，CPU 负责将外部状态继电器的状态通过采集电路转换成采集信息，经过处理后再通过 CAN 总线向联锁机发送。在驱动模块中，CPU 通过 CAN 总线接收来自联锁机的控制命令，经过双 CPU 比较后，同时产生驱动信号。

另外，CPU 的另一个主要功能是完成模块的自检。模块在每个正常的工作周期中都要运行自检程序，进行故障检测及发现故障后的故障定位和报警，以此提高模块的安全性和可靠性。

CPU 板由 CPU、SRAM、FLASH、FIFO、FPGA、CAN 控制器和电源管理单元组成，如图 4.1.1 所示。

用 FPGA 实现 CPU 的外围译码逻辑、硬件同步电路、两路同步串口（用于与其它两块 CPU 板通讯）、两路异步串口（用于 VxWorks 系统调试）、监视电路（用于对 3 块 CPU 板运行状态的进行监视）以及看门狗等功能。

监测与电源管理单元对 CPU、RAM、FLASH、电源电压（24V，5V）进行监测；根据监测结果或外部信号，接通或切断 CPU 板的电源。当某块 CPU 板发生故障时，可通过上位机、自身或其它 2 块 CPU 板切断电源，等待维修。

CAN 控制器用于与 I/O 接口板通讯。CAN 总线控制器采用 PHILIPS 公司的 SJA1000，支持热插拔（Hot Plugging Support）。硬启动时，芯片进入复位模式；通过初始化，可以使 SJA1000 脱离复位状态。每块 CPU 板有一路 CAN 总线接到 I/O 接口系统的底板上。

8M 字节的 FLASH 用于固化程序，8M 字节的 SRAM 用于存储数据。FIFO 用于 CPU 板与通讯板交换数据。

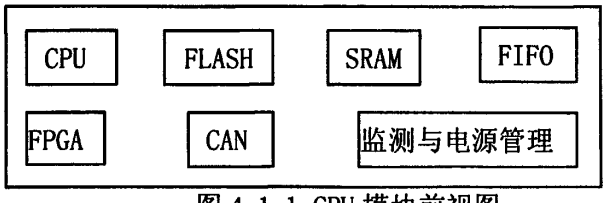

图 4.1.1 CPU 模块前视图

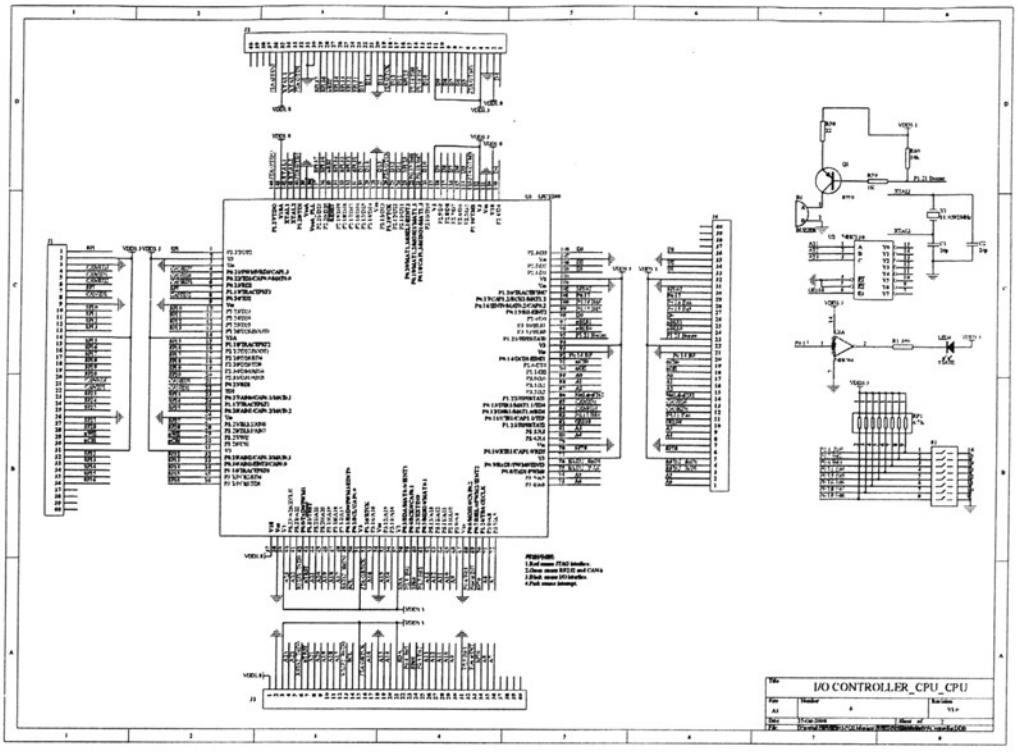

图 4.1.2 CPU 模块上的单个 CPU 原理图

图 4.1.2 为 CPU 模块上 CPU 的原理，CPU 外围的四个插座为调试插座，不焊接实际的插座，只为调试用，右边为蜂鸣器和预留的译码电路。

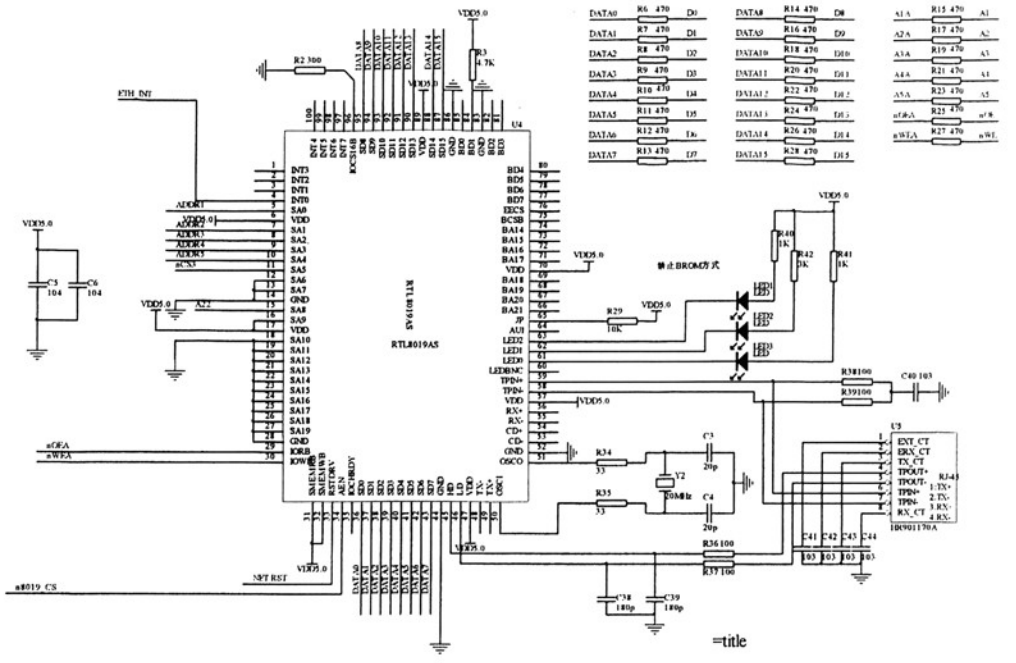

图 4.1.3 以太网接口图

图 4.1.3 为以太网电路，以太网芯片采用 RTL8109，RTL8109 是成熟的通用以太网芯片，插座采用带有防雷保护的 RJ45 插座。

CPU 板的 PCB 图见附录 A.1。

4.2 CAN 总线通信部分

LPC2294 自带须有 4 个 CAN 控制器，配上外部的驱动芯片 TJA1050 很容易完成对 CAN 的控制。

CAN 控制器主要功能是将要发送的数据进行处理，使其适合在总线上发送。同时对总线上的数据进行判别，接收属于自己的信息。CAN 控制器主要由接口管理逻辑（IML）、发送缓冲器（TXB）、接收缓冲器（RXB，RXFIFO）、接收过滤器（ACF）、位流处理器（BSP）、位时序逻辑（BTL）和错误管理逻辑（EML）等几部分构成，其中发送缓冲器长 13 个字节，由 CPU 写入、BSP 读出。

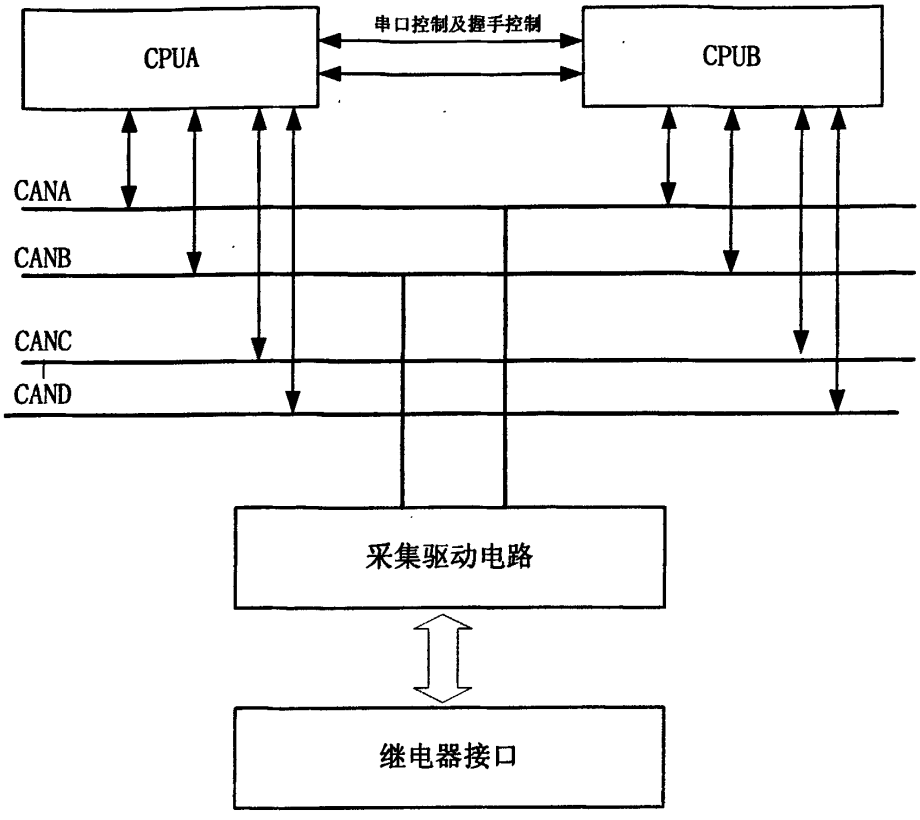

图4.2.1　智能I/O模块通信原理图

图4.2.1为智能I/O模块通信原理图，当被配置成2取2模式时，二个CPU模块通过串口交换数据，通过双路CANA，CANB和采集驱动电路通信，和外部的连接通过CANC，CAND通信。

当I/O控制器被配置成2×2取2模式时，要配成2个机框；当I/O控制器配置为3取2模式时，只需要一个主机框，在主机框内安装两块通讯板，作为双板热备。主通讯板负责与外界通讯；备板的外部数据接口为高阻状态。主、备板之间的切换可由CPU板根据通讯板返回的状态信息、上位机的指令、或系统定时进行。

当安全计算机配置为2乘2取2模式时，主机部分为两个机框，每个机框内安装1块通讯板。通讯板的主、备属性切换可由上位机指令配置、或系统定时进行。

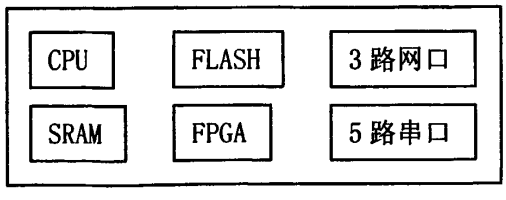

图4.2.2 通信板结构图

通讯板由 CPU、FPGA、3 路网口和 5 路串口组成，如图 4.2.2 所示。CPU 负责对网口和串口初始化、读写各 CPU 板 FIFO 的数据，发送和接收各串口和网口的数据。FPGA 实现的功能如下：当安全计算机配置为 3 取 2 工作模式时，FPGA 对从 3 个 CPU 的 FIFO 发来的数据进行 3 取 2；当安全计算机配置为 2 乘 2 取 2 时，FPGA 对从 2 个 CPU 的 FIFO 发来的数据进行 2 取 2。FPGA 还根据数据的标志位，配合 CPU 将此数据发往相应的网口或串口；将从各网口和串口的接收到的数据添加适当的标志位，发往各 CPU 板。各路网口和串口均为独立模块。

网口用于与上位机和维修机通讯。网卡采用 82551 芯片，不必自己开发网卡驱动。

串口用于与其它现场设备连接。

4.3 输入输出模块

输入输出板采用 ARM2294 芯片，工作频率为 24MHz,完成以下功能：

1) 32 点输入或输出；
2) 和 CPU 模块进行通讯；
3) 对输入输出进行在线自检；
4) 人工设置输入和输出；
5) 将工作状态信息和错误报告及时传送给 CPU 模块；
6) 对输入进行滤波，滤波时间分为 1,5，10,,50,100ms；
7) 前面板 LED 动态显示电源、通讯和输入输出状态；
8) 对输入和输出进行光耦隔离；
9) 光电隔离器件两测电源完全分开供电；

参数：输入输出模块的框图如下：

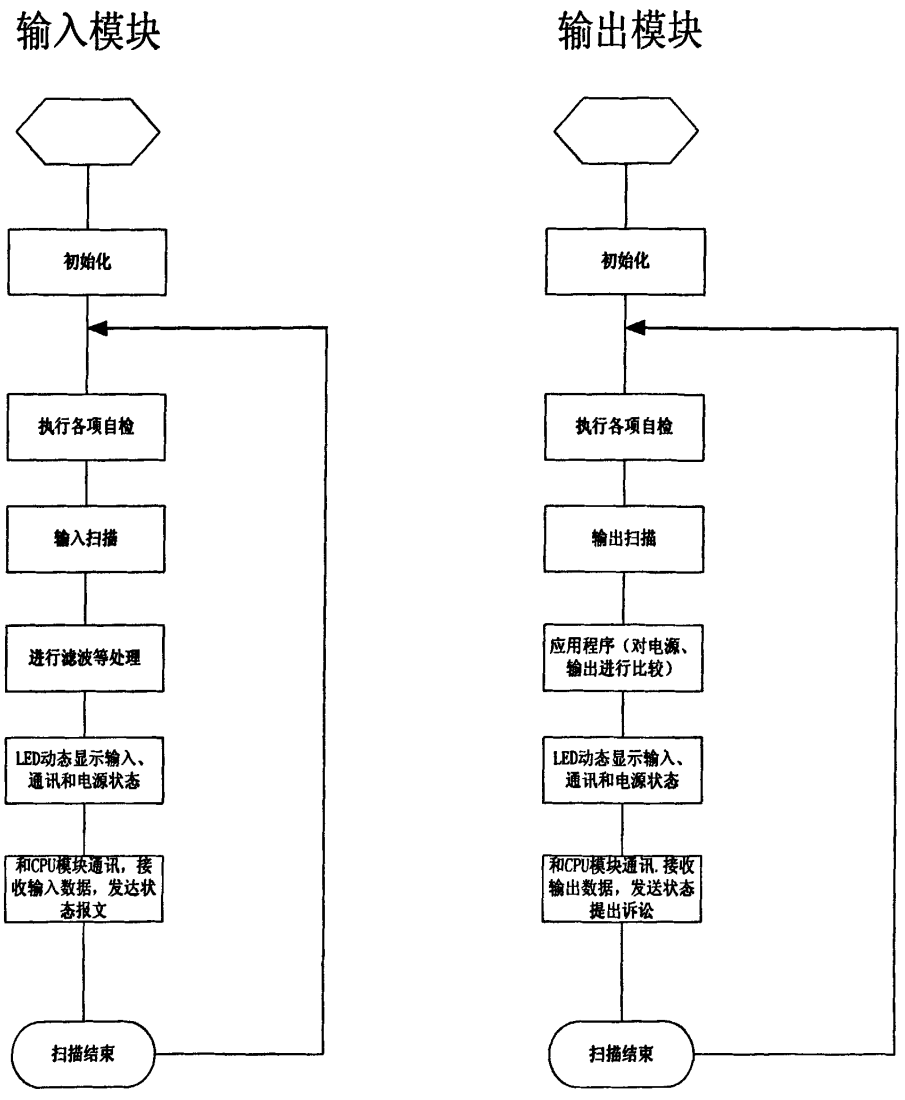

图 4.3.1 输入、输出模块框图

图 4.3.1 中表明了输入输出模块的主调度图，对于输入模块来讲，上电后，先对所有硬件和软件进行初始化，包括加载数据配置文件，然后进行各项自检，包括芯片自检，内存的读写自检，配置文件正确性检查，模块状态检测等，完成后对输入状态进行扫描，为防继电器抖动造成的输入状态变化，采用滤波方法，输入状态扫描完成后对面板 LED 的状态灯进行显示，显示完成后对 CPU 模块进行通讯，把采集状态通过双通道传送给 CPU 模块。

技术数据：

1) 输入电压：5V，24V；

2) 每路电流最大 100mA；

3) 环境温度：0 度-70 度；

4) 绝缘强度：2kVeff(有效值)；

5) CAN 总线最大连接长度：100m。

I/O 板每两块一组，构成热备冗余通道。当 I/O 板配置为备份的输入接口板时，该板不向各 CPU 板发送预处理后的输入开关量的值；当 I/O 板配置为备份的输出接口板时，该板不向外界输出开关量，输出端口为高阻。

每块 I/O 板由 4 路 CAN 总线接口、两片 FPGA、CPU、光耦隔离等单元组成，如图 4.3.2 所示。2 路 CAN 用于与各 CPU 板通讯。

可通过 I/O 板上的开关，手动配置成输入接口板或输出接口板。上电复位后或初始化时，CPU 根据读入的配置开关值，确定此板为输入接口板还是输出接口板。I/O 接口板的前面板安装发光二极管，对每路输入、输出状态进行动态显示。

当 I/O 板配置为输入接口板时，工作过程如下：输入开关量首先经过两路并行的光耦隔离后，分别送到两片 FPGA 进行滤波和鉴别（区分有效的开关量信号与 24V 电源电压，以确定是否是信号线与电源线搭接）。然后在 CPU 内，对两片 FPGA 的输出结果进行比较，若比较结果一致，将此值经 CAN 总线送到各 CPU 板处理；若比较结果不一致，在 I/O 板接口板的前面板点亮一个故障指示灯，同时经过各 CPU 板向上位机发送信息，报告错误，以便维修或切换到处于热备状态的对应 I/O 接口板。CPU 还用于初始化和管理本板的 CAN 总线节点。

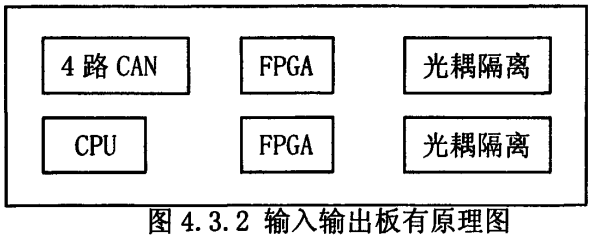

图 4.3.2 输入输出板有原理图

当 I/O 板配置为输出接口板时，工作过程如下：当安全计算机工作于 3 取 2 模式时，每块 FPGA 实现输出开关量的 3 取 2 逻辑；当工作于 2 乘 2 取 2 模式时，每块 FPGA

实现输出开关量的 2 取 2 逻辑。两片 FPGA 的输出分别经光耦隔离后，控制安全继电器的两条控制线，实现安全输出。同时输出信号再经光耦隔离后，供 CPU 回读。回读结果经 CAN 总线送到各 CPU 板。CPU 还用于管理本板 CAN 总线节点和回读输出的开关量。

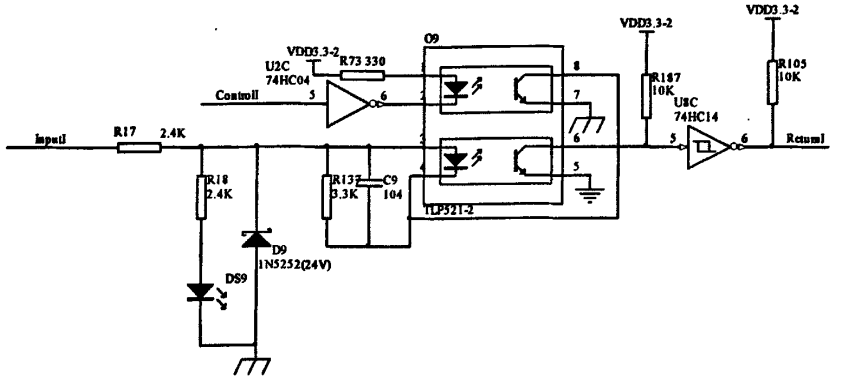

图 4.3.3 采集输入原理图

图 4.3.3 为采集输入的原理图，U2C 的输入信号来自输入模块的 CPU，假设为低电平 1，则输出 U2C 的输出管脚 6 为 0，发光管 12 导通，InputI 为继电器接点输入，当继电器接点导通是时，从外部输入 24V 直流电压，发光管 34 导通，这时由于发光管 12 导通过，所以 U8C 的输入脚 5 为 0 ，ReturnI 返回 1,当发光管的 12 截止时，U8C 输出 0。综上所述，当 CONTROLI 输入为方波信号时，如果继电器接点闭合，在 ReturnI 端将得到反相的动态信号；当继电器接点断开时，在 ReturnI 端一直是 0 电平。

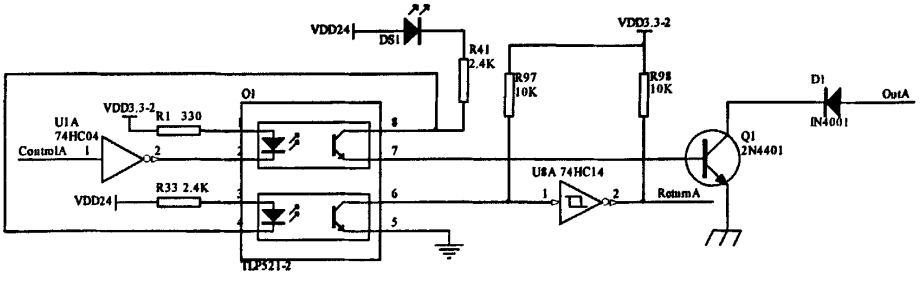

图 4.3.4　输出模块原理图

图 4..3.4 为输出的原理图，图中只给出一路输出，其中 ControlA 信号来输出模块的 CPU，当输出为高电平 1 时，U1A 的 2 脚输出 0，发光管的 12 ，Q1 导通，同时发光管 34 导通，返回信号 ReturnA 为 1，当控制信号 ControlA 为 0 时，发光管 12

截止，Q1 截止。

输入板的 PCB 图见附录 A.2，输出板的 PCB 图见附录 A.3。

4.4 继电器板

为方便调试，用来模拟外部安全继电器。

技术数据：

1) 每块板 32 个输入，32 输出；

2) 通过跳线改变输入、输出方式；

3) 32 路线圈吸合指示。

继电器板将每一个继电器的每一个接点都分别引到不同的插座上供调试用，通过跳线来实现一个板即可以用作输入模块的外部继电器调试板，又可以用作输出模块的外部继电器调试板。

继电器板的 PCB 图见附录 A.4。

4.5 采集部分故障-安全实现

采集模块的输入部分采用 CPU 与硬件电路配合，形成“异或”逻辑，既保证了采集功能，又提高了故障的检出率和检出速度[35]。采集部分的原理见下图。

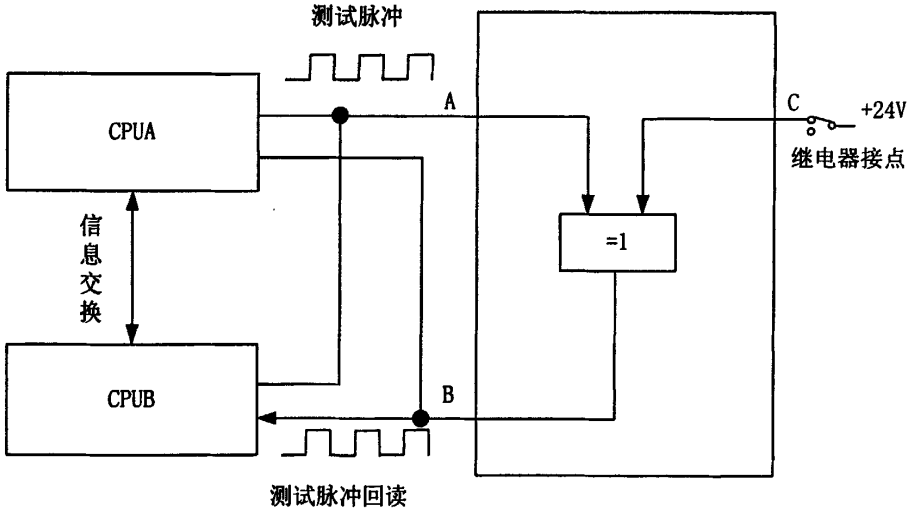

图4.5.1 采集部份原理

图中 A 点为测试用的驱动脉冲；B 点为测试脉冲的回读；c 点为外部设备继电器的接点状态。它们三者经过逻辑和隔离部分的转换后满足“异或”逻辑关系。

在电路正常工作的情况下，外部继电器的接点状态（C 点的值）是由 a、b 两点

经过“异或”逻辑计算所得，并且只要在A点输入测试脉冲，B点都会有相应的回读脉冲，这也达到了循环自检的目的。如果逻辑电路部分发生故障，则B点回读信号为稳定的电平，由于CPU 部分在每个周期都要进行自检，所以故障能被及时检测出来，使模块能及时报警。

4.6 驱动部份故障-安全实现

驱动模块的输出部分采用双CPU比较直流输出的闭环结构。其原理见下图。

从下图中可看出，在驱动模块接收到驱动命令后，首先由CPU A和CPU B进行处理和比较，经比较一致后共同发出控制信息，驱动外部继电器J。另外每个CPU的驱动信息都有回读信息，从而构成了闭环检测。另外在运行周期中，对没有输出的分路进行自检，防止这些分路的故障被隐蔽。通过这种方法可以保证模块的任何输出分路，不管它的使用频度如何，一旦发生故障，均能被及时检测出来，并得到相应处理。

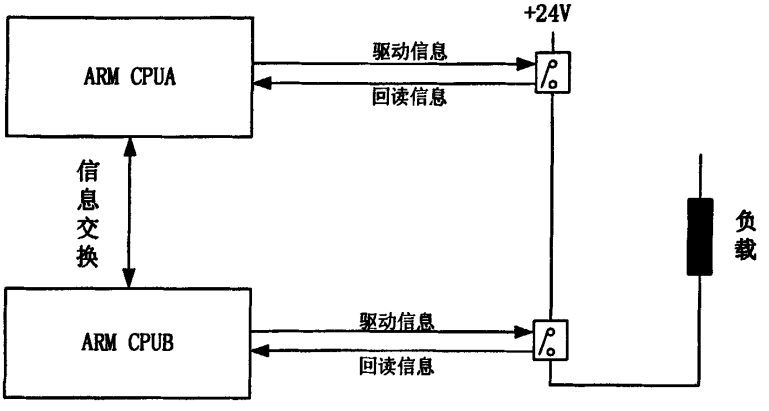

图4.6.1　驱动原理

4.7 电源部分

模块内除正常的芯片电源外，在驱动模块中增加了故障-安全的鉴相驱动电源。加入这个电源的目的是在模块发生危险侧故障时切断驱动电源，使模块导向安全侧。

5 软件结构

5.1 uC/OS-II 简介

uC/OS-II 是一个简单、高效的嵌入式实时操作系统内核，被应用到各种嵌入式系统中。目前，它支持 x86、ARM、PowerPC、MIPS 等众多体系结构，并有上百个商业应用实例，其稳定性和可用性是经过实践验证的。同时，它的源代码公开，可以从 www.ucos-ii.com 网站上获得全部源码以及其在各种体系结构平台上的移植范例 [36]。

最新的 uC/OS-II 2.0 版以上的内核都具有可抢占的实时多任务调度功能，另外它还提供了许多系统服务，例如信号量、消息队列、邮箱、内存管理、时间函数等，这些功能可以根据不同的需求进行裁减。可以说，uC/OS-II 是一个具备现代操作系统特点的 RTOS，同时它结构清晰、注解详尽，具有良好的可扩展性和可移植性，被广泛地应用于各种架构的微处理器上。

5.2 ARM 的移植

在开发嵌入式系统时,一般选择基于 ARM 和 uC/ OS - II 的嵌入式开发平台,因为 ARM 微处理器具有处理速度快、超低功耗、价格低廉、应用前景广泛等优点。 将 uC/ OS - II 移植到 ARM 系统之后,可以充分结合两者的优势. 如果一个程序在一个环境里能工作,我们经常希望能将它移植到另一个编译系统、处理器或者操作系统上,这就是移植技术.移植技术可以使一种特定的技术在更加广泛的范围使用,使软件使用更加灵活,不局限于某一条件.uC/OS - II 是由 Jean J . Labrosse 先生编写的完整的可移植、固化、裁剪的占先式实时多任务内核.uC/ OS - II 的源代码完全开放,这是其他商业实时内核无法比拟的[2] . 它是针对嵌入式应用设计的,在设计之初就充分考虑了可移植性,它的大部分源代码都是用高可移植性的 ANSIC 编写的。uC/ OS - II 可以移植到从 8 位到 64 位的不同类型、不同规模的嵌入式系统,并能在大部分的 8 位、16 位、32 位、甚至 64 位的微处理器和 DSP 上运行. 由于 uC/ OS - II 是一个实时操作系统,所以如果将它嵌入到 ARM 处理器上,就能够进一步简化 ARM 系统的开发.

轨道交通运行控制系统大多是实时、多任务和安全苛求的计算机控制系统，

uC/OS - II 的文件系统结构包括核心代码部分、设置代码部分、与处理器相关的移植代码部分，其中最上边的软件应用层是 uC/ OS - II 上的代码. 核心代码

部分包括7个源代码文件和1个头文件. 功能分别是内核管理、事件管理、消息队列管理、存储管理、消息管理、信号量处理、任务调度和定时管理. 设置代码部分包括2个头文件,用来配置事件控制块的数目以及是否包含消息管理相关代码. 而与处理器相关的移植代码部分则是进行移植过程中需要更改的部分,包括1个头文件OS CPU. H,1个汇编文件OS CPU A. S和1个C代码文件.实际上将uC/ OS - II移植到ARM处理器上,需要完成的工作主要是以下三个与体系结构相关的文件:OS CPU. H,OS CPU. C以及OS CPU A[37]。

5.2.1 OS CPU. H的移植

在将uC/ OS - II移植到ARM处理器上时,首先进行基本配置和数据类型定义. 重新定义数据类型是为了增加代码的可移植性,因为不同的编译器所提供的同一数据类型的数据长度并不相同,例如int型,在有的编译器中是16位,而在另外一些编译器中则是32位. 所以,为了便于移植,需要重新定义数据类型,如INT32U代表无符号32位整型. typedefunsigned int INT8U,就是定义一个8位的无符号整型数据类型. 其次就是对ARM处理器相关宏进行定义,如ARM处理器中的退出临界区和进入临界区的宏定义,退出临界区宏定义: # define OS EXITCRITICAL () ARMDisable Int () / / 关中断,进入临界区宏定义# define OS ENTER CRITICAL ( ) AR2MEnableInt () / / 开中断. 最后就是堆栈增长方向的设定. 当进行函数调用时,入口参数和返回地址一般都会保存在当前任务的堆栈中,编译器的编译选项和由此生成的堆栈指令就会决定堆栈的增长方向[6],定义为# define OS STK GROWTH 1.

5.2.2 OS CPU. C的移植

OS CPU. C的移植包括任务堆栈初始化和相应函数的实现. 在这里,共有6个函数:OSTaskStkInit( ), OSSTaskCreateHook ( ), OSTaskDelHook ( ), OS2Task SwHook( ),OSTaskStatHook ( ), OSTimeTickHook () . 其中后面的5个HOOK函数又称为钩子函数,主要是用来对uC/ OS - II进行功能扩展. 这些函数为用户定义函数,由操作系统调用相应的HOOK函数去执行,在一般情况下,他们都没有代码,所以实现为空函数即可. 而函数OSTaskStkInit ( )对堆栈进行初始化,在ARM系统

中,任务堆栈空间由高到低依次为 PC ,LR ,R12 ,R11 , …,R1 ,R0 ,CPSR ,SPSR. 在进行堆栈初始化以后,OSTaskStkInit ( ) 返回新的堆栈栈顶指针.

5.2.3 OS CPU A. S 的移植

OS CPU A. S 文件的移植需要对处理器的寄存器进行操作,所以必须用汇编语言来编写. 这个文件的实现集中体现了所要移植到处理器的体系结构和 uC/ OS - II 的移植原理[6] . 它包括 4 个子函数:OSStartHighRdy() , OSCtxSw() , OSIntCtxSw() ,OSTick2ISR() . 其中难点在于 OSIntCtxSw() 和 OSTickISR() 函数的实现,因为这两个函数的实现与移植者的移植思路以及相关硬件定时器、中断寄存器的设置有关.在实际的移植工作中,这两处也是比较容易出错的地方.

OSIntCtxSw( ) 函数由 OSIntExit ( ) 函数调用,而 OSIntExit () 函数又由 OSTickISR() 调用. OSIntCtxSw()函数最重要的作用就是它完成在中断 ISR 中直接进行任务切换,从而提高了实时响应的速度. 它发生的时机是在 ISR 执行到 OSIntExit ( ) 时,如果发现有高优先级的任务因为等待 time tick 的到来获得了执行:uC/ OS - II 在 ARM 系统上的移植与实现的条件,就可以马上被调度执行,而不用返回被中断的那个任务之后再进行任务切换. 实现 OSIntCtxSw() 的方法大致也有两种情况[7] :一是通过调整 SP 堆栈指针的方法,根据所用的编译器对于函数嵌套的处理,通过精确计算出所需要调整的 SP 位置来使得进入中断时所作的保护现场的工作可以被重用. 二是设置需要切换标志位的方法,在 OSIntCtxSw( ) 里面不发生切换,而是设置一个需要切换的标志,等函数嵌套从进入 OSIntExit ( ) = > OS ENTER CRITI2CAL() = > OSIntCtxSw( ) = > OS EXIT CRITICAL() = > OSIntExit ( ) 退出后,再根据标志位来判断是否需要进行中断级的任务切换.

其次是对 OSTickISR() 修改.OSTickISR() 首先在被中断任务堆栈中保存 CPU 寄存器的值,然后调用 OSIntEnter () . 随后调用 OSTimeTick() ,检查所有处于延时等待状态的任务,判断是否有延时结束就绪的任务. 最后调用 OSIntExit ( ) . 如果在中断中(或其他嵌套的中断) 有更高优先级的任务就绪,并且当前中断为中断嵌套的最后一层,OSIntExit ( ) 将进行任务调度. 如果进行了任务调度,OSIntExit ()

将不再返回调用者,而是用新任务的堆栈中的寄存器数值恢复CPU 现场,然后实现任务切换. 如果当前中断不是中断嵌套的最后一层,或中断中没有改变任务的就绪状态,OSIntExit ( ) 将返回调用者OSTickISR ( ) ,OSTickISR() 返回被中断的任务. 最后就是退出临界区和进入临界区函数. 进入临界区时,必须关闭中断,用ARMDisableInt () 函数实现. 在退出临界区的时候恢复原来的中断状态,通过ARMEnableInt ( ) 函数来实现[7] . 至于进行任务级上下文切换,则是由汇编子程序OSCtxSw 实现.

uC/ OS - II 的多任务机制非常灵活，具有任务优先级别，任务优先级的确定要综合考虑任务的重要程度、运行时间以及触发频率等因素。对于安全计算机软件来说，我们按照欧洲铁路标准EN61508《电气/电子/可编程电子安全相关系统的功能安全》的思路对其任务处理优先级规划如下：安全苛求功能(Safety Critical Function)运算处理应具有较高优先级，例如计算机联锁中的进路处理任务或道岔单操控制任务或列车自动防护中的制动速度曲线计算任务等；安全相关功能（Safety-related Function）运算处理具有次高优先级，例如输入信息采集任务或主处理单元间通信校验任务等；非安全相关功能任务具有较低的优先级；而硬件中断级任务，如通信中断等应享有最高优先级，以保证其高度实时性。

轨道交通信号系统的核心是安全计算机，对它的研究具有重要的理论和实践意义。本方案设计了基于ARM 的冗余结构安全型输入输出，实现了主处理单元的uC/ OS - II 底层驱动程序，并且进行了长时间的多任务负荷老化实验，运行稳定可靠。

5.3 对软件安全性的要求

风险可被定量化，但不能以同样方式规定软件安全完整性。因此，对于本欧洲标准，软件安全完整性等级被指定为下列五个等级之一：[38]

软件安全完整性等级	软件安全完整性等级描述
4	非常高
3	高
2	中等
1	低
0	非安全性

按照欧洲标准要求，信号产品的软件完整性等级要求达到最高的4级，软件开发围绕产品的整个生命开发周期，图5.3.1为软件标准要求的整个开发生命周期V字型模型。

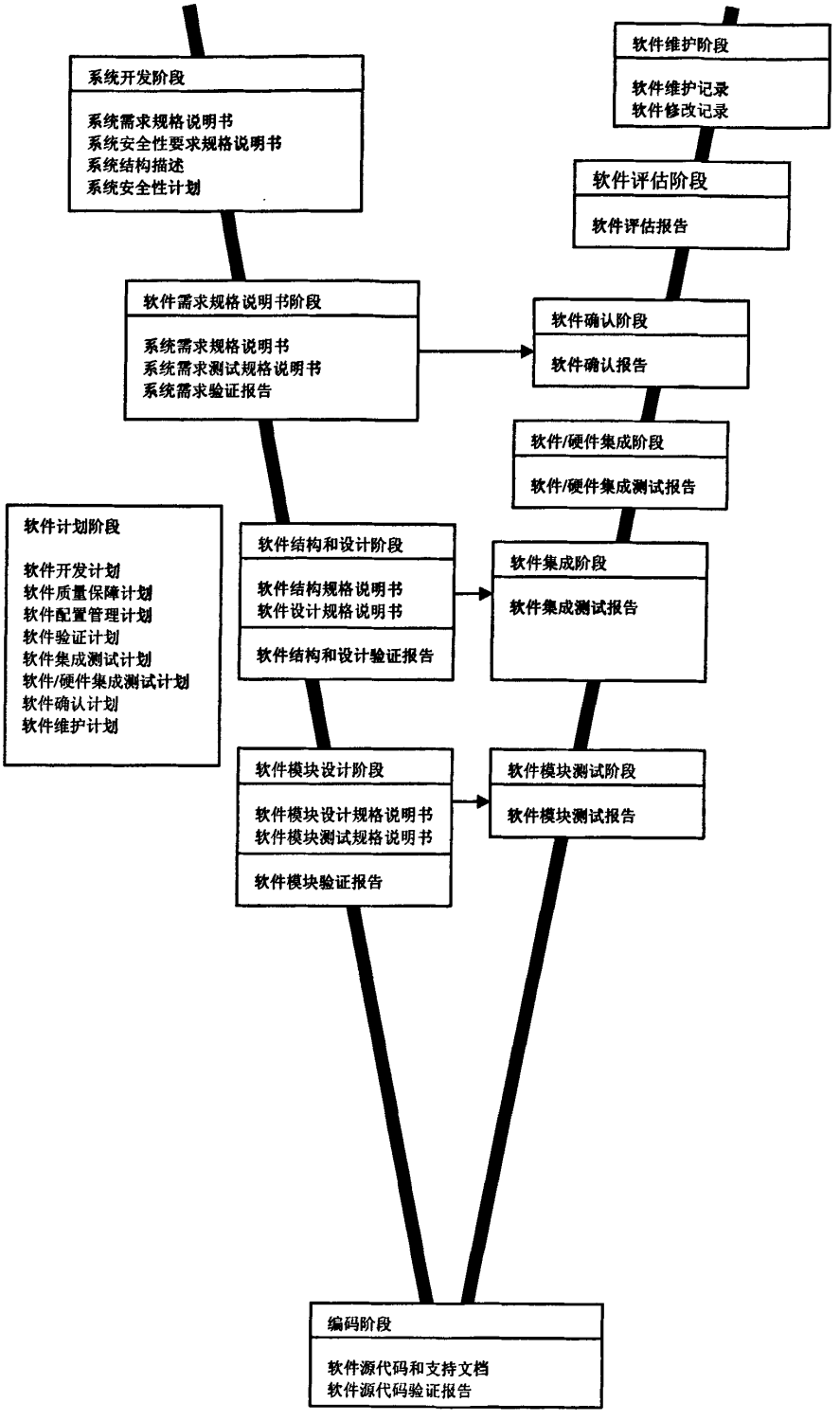
系统开发阶段
系统需求规格说明书
系统安全性要求规格说明书
系统结构描述
系统安全性计划
软件维护阶段
软件维护记录
软件修改记录
软件评估阶段
软件评估报告
软件需求规格说明书阶段
系统需求规格说明书
系统需求测试规格说明书
系统需求验证报告
软件确认阶段
软件确认报告
软件/硬件集成阶段
软件/硬件集成测试报告
软件计划阶段
软件开发计划
软件质量保障计划
软件配置管理计划
软件验证计划
软件集成测试计划
软件/硬件集成测试计划
软件确认计划
软件维护计划
软件结构和设计阶段
软件结构规格说明书
软件设计规格说明书
软件结构和设计验证报告
软件集成阶段
软件集成测试报告
软件模块设计阶段
软件模块设计规格说明书
软件模块测试规格说明书
软件模块验证报告
软件模块测试阶段
软件模块测试报告
编码阶段
软件源代码和支持文档
软件源代码验证报告

图 5.3.1　开发生命周期

图 5.3.1 发信号产品的软件开发生命周期图，从图中可以看出左边为开发的每个阶段所要完成的任务和递交的文件，右边为测试和验证要完成的工作，左边的工作内容做为右边的输入。如需求阶段，需要完成系统需求规格说明书，系统测试规格说明书和系统需求验证报告，这些文件作为软件确认阶段时产生的确认报告的输入。

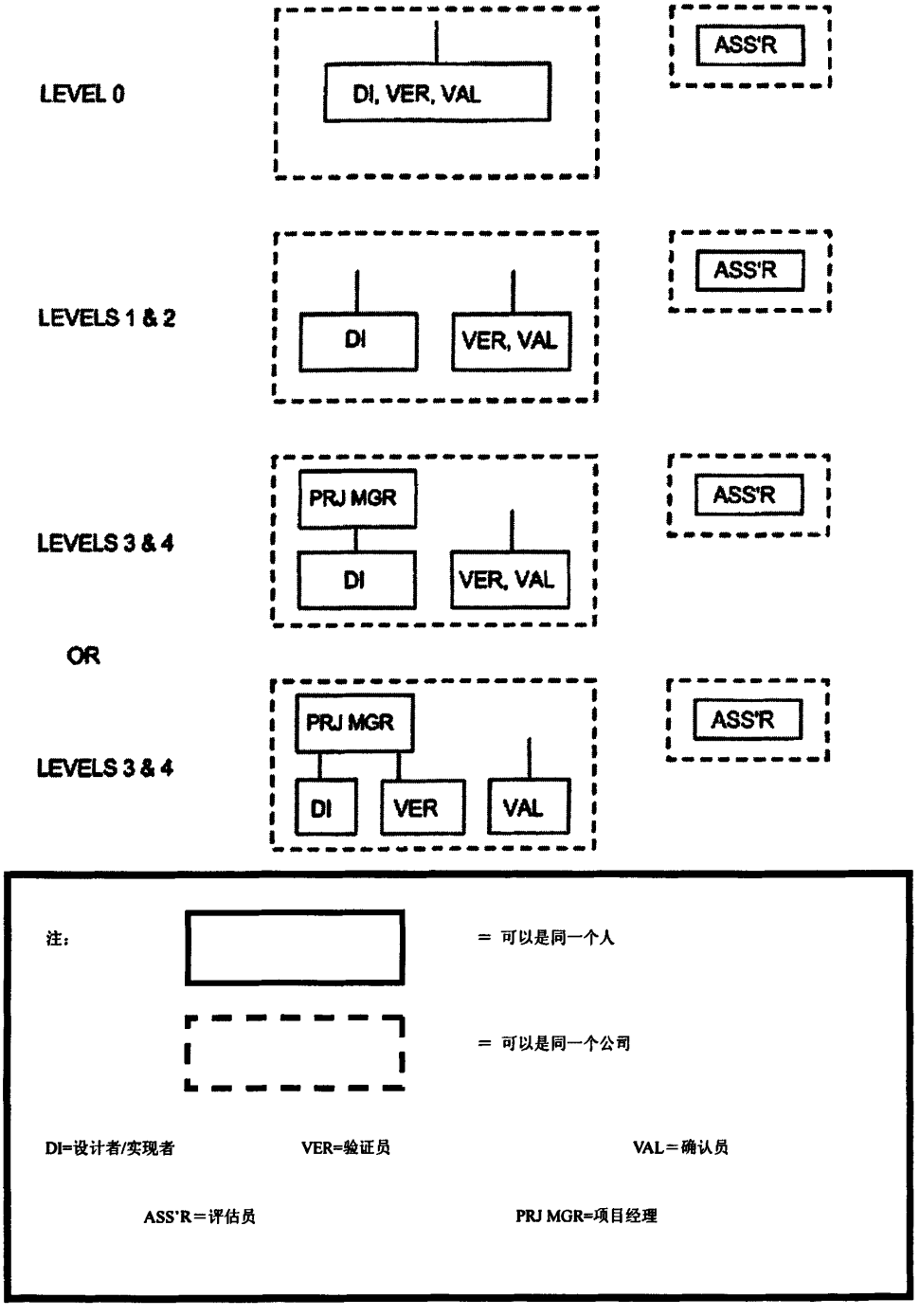

图 5.3.2 开发生命周期中的人员组织

图 5.3.2 为项目开发过程中的人员组织安排，从图中可以看出，对于 4 级要求主评估员要求独立于项目团队，同时在项目团队中确认员验证人员和设计人员必须分开，确认员不能服从于项目经理的领导。

5.4 软件设计

5.4.1 安全冗余管理中间件

安全型智能输入输出的应用开发只是输入的采用和对输出的控制，因而比较简单，系统软件的核心在冗余管理中间件部分，它要把离散的单机组成一个统一的系统，对于应用来说不用关心系统的硬件。

安全冗余管理中间件部分是安全型输入输出的核心，其层次图如下图。

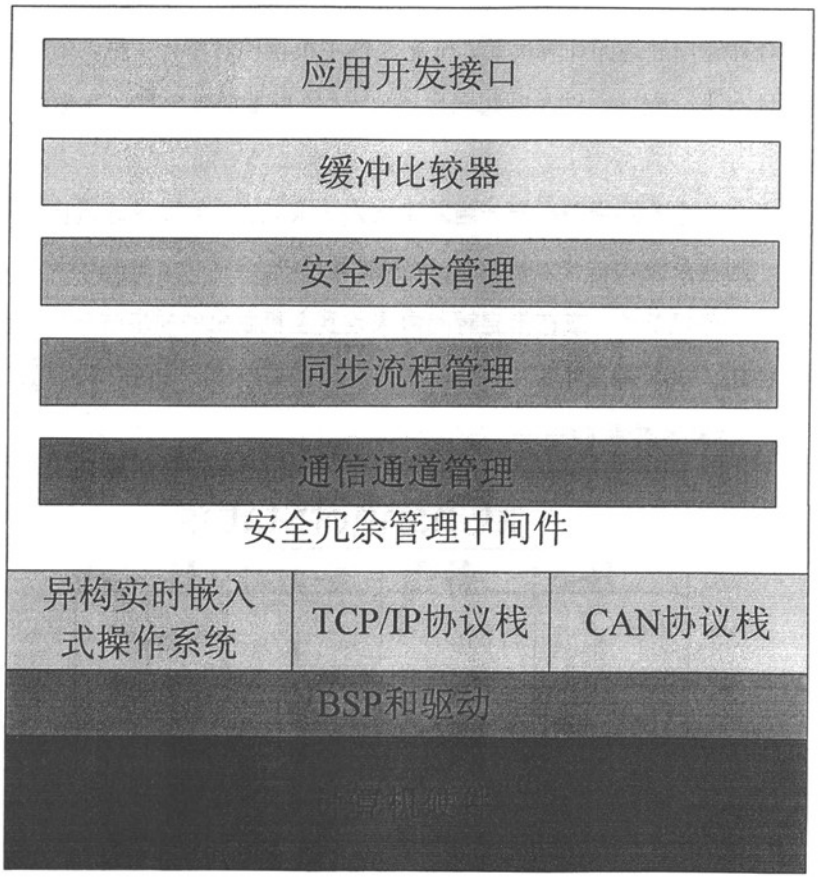

图 5.4.1　系统层次结构图

冗余管理中间伯大致分为通信通道管理、同步流程管理、安全冗余管理、缓冲比较器和应用开发接口五个子模块。如图 5.4.1 所示。其中最底层为计算机硬件，其上为 BSP 和驱动，再上层为操作系统和协议栈，再上面为安全冗余管理中间件。

安全智能型 I/O 模块的软件采用结构化设计方案，这种方案具有良好的模块化，可修改性及可移植性。软件应用 C 语言进行编写，具有较高的效率，可读性好，易于修改。模块的软件设计主要包括初始化、通信处理、采集驱动命令处理、自检处理 4 个部分。

为保证模块的可靠性，防止模块的程序在运行过程中“走飞”而产生危险侧输出，模块中设有看门狗定时器。由于模块采集、驱动以及自检功能都是由 CPU 的正常工作来保证，如模块的 CPU 发生故障，则看门狗定时器起作用，使模块导向安全侧，从而保证模块的安全性。

5.4.2 冗余实现

当安全型输入输出被配置为三取二时，我们采取的三取二原则如下：

1) 存在对外输出通道的计算单元、优先获得主机权限者作为主机。由主机对三机系统的状态进行配置，以及发起同步(每个运算周期同步一次，在应用任务开始之前)。
2) 主机每个状态检测周期检查一遍通讯状态记录表。一旦发现通讯状态记录表发生变化，则将系统状态设为初始态，并重配置系统。
3) 从机满足停机条件，或在指定时间内未收到主机发出的同步信号则停机。（所有任务关闭，包括通讯任务

通讯状态记录表如下：

aChannelStatus			
CH-a	备用	S-a(a)	H-a(a)
1		1	0
CH-b(b)	CH-b(c)	S-b(b)	H-b(b)
0	0	0	0
CH-c(c)	CH-c(b)	S-c(c)	H-c(c)
0	0	0	0

图 5.4.2 通讯状态记录表

图中的通讯状态记录表是以 A 机为例的，第一行：系统自身状态(a 单元)，CH-a(a)：

a 自身监测到的通道状态，S-a(a)：a 自身系统状态，H-a(a)：a 自身主机状态，第二行：第一个伙伴通道状态(b 单元) CH-b(b)：b 自身监测到的通道状态，CH-b(c)：从 c 发来的 b 单元通道状态，S-b(b)：b 自身系统状态，H-b(b)：b 自身主机状态；第三行：第二个伙伴通道状态(c 单元)，CH-c(c)：c 自身监测到的通道状态，CH-c(b)：从 b 发来的 c 单元通道状态，S-c(c)：c 自身系统状态，H-c(c)：c 自身主机状态。

B 和 C 机也有同样的状态表，通过对三机的状态判别，从而使系统完成三取二的冗余。

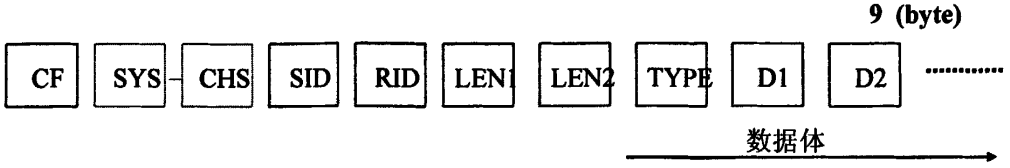

图 5.4.3 报文结构

图 5.4.3 中的各项说明如下：

CF ：控制域

SYS：当前单元所处系统状态以及主机状态

CHS：当前单元通道状态以及获得的另一伙伴通道状态

SID：发出的报文 ID 号

RID：接收到的报文 ID 号

LEN1：数据长度(高位)

LEN2：数据长度(低位)

TYPE：数据类型(标识数据报文或者指令报文)

D1,D2,..... 数据体

5.4.3 主程序主要框图如下：

对于冗余内核管理程序（KS），按照软件工程化的要求，设计主要说明文件如下：

表 5.4.1 KS 中包含的一级 CSC 汇总表

序号	一级 CSC 名称	CSC 标识	CSC 用途	下一级 CSC(或 CSU)名称
1	通道初始化	C_KS_INIT	完成计算通道的 KS 软件初始化	无
2	通道管理	C_KS_GOV	对本通道进行相对独立的管理	报警及记录管理、缓冲管理
3	通道间信息管理	C_KS_ADM	管理通道间的交互，实现主要的三取二功能	通道间通信、通道间状态检测、时钟同步、通道切换及通道关闭、三取二表决、通道重配置
4	输入输出管理	C_KS_IO	完成应用数据的输入、分发及输出功能	数据输入、数据输出

5	自诊	C_KS_SD	系统自诊断	无

如上表所示，将 I_KS 划分为 5 个一级 CSC，为分别为通道初始化、通道管理、通道间信息管理、输入输出管理及自诊。其中通道初始化负责在计算通道启动时对其进行所需的全部初始化工作，为其它 CSC 的运行作准备；通道管理主要负责对只涉及本通道的事务进行管理，主要提供通道缓冲管理及为其它 CSC 提供报警及记录管理；通道间信息管理 CSC 主要完成涉及到三取二系统的核心功能，维护整个系统的正常运行；输入输出管理 CSC 只负责对应用数据的输入输出管理工作，但系统是否进行输入输出及如何输入输出取决于通道间信息管理的结果；自诊功能主要独立地对系统硬件进行诊断。

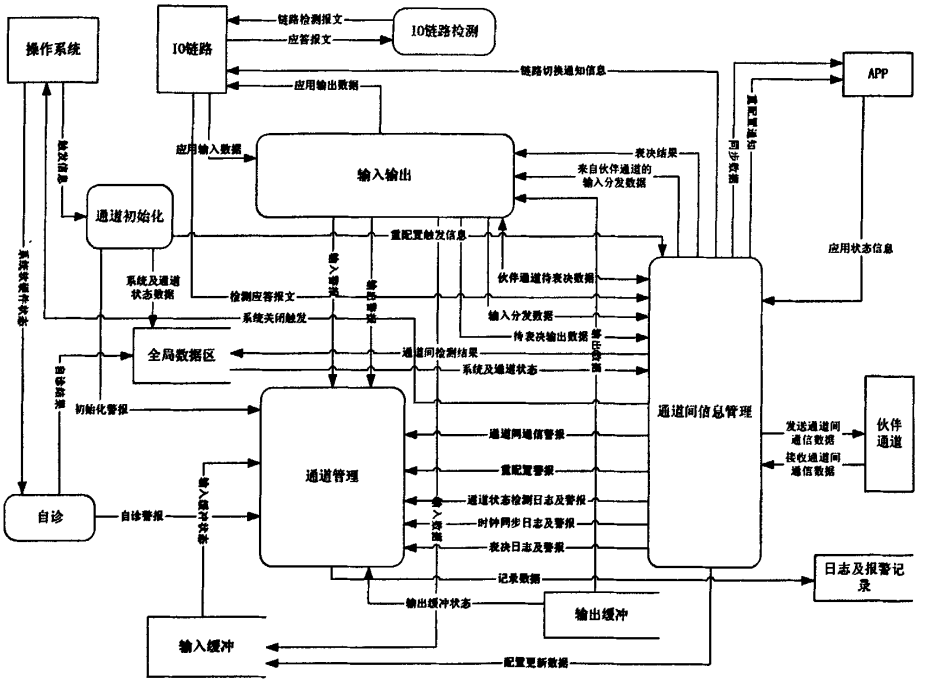

图 KS顶层数据流图

图 5.4.4 一级 CSC 之间接口关系

图 5.4.4 对五个一级 CSC 之间的接口关系进行了说明。

对于通道初始化，无二级 CSC。

表 5.4.2 通道管理二级 CSC

序号	二级 CSC 名称	CSC 标识	CSC 用途	下一级 CSC 名称
1	缓冲管理	C_KS_GOV_BUF	对 KS 提供的输入输出缓冲进行管理	无
2	报警及记录	C_KS_GOV_LA	重要信息的记录及报警管理	报警、记录

表 5.4.2 对通道管理的二个二级 CSC 进行了说明。

通信通道管理 CSC 结构

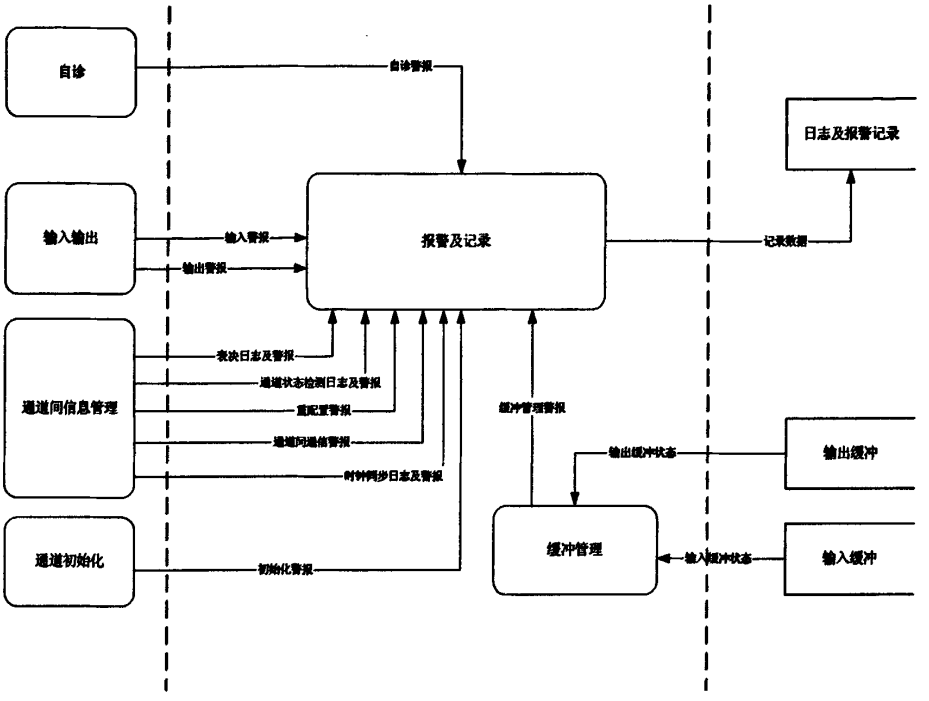

图 KS通道管理二级数据流图

图 5.4.5 通道管理二级 CSC 接口关系

图 5.4.5 为输入输出二级 CSC 的接口管理以及相互间的接口关系

对每个一级 CSC 进行描述。只画到下属 CSC

表 5.4.3　通道间信息管理二级 CSC

序号	二级 CSC 名称	CSC 标识	CSC 用途	下一级 CSC 名称
1	通道间通信	C_KS_ADM_COM	提供三个通道间的通信服务	无
2	通道间状态检测	C_KS_ADM_DET	维护通道间的心跳信息	无
3	时钟同步	C_KS_ADM_TIM	实现三个通道间的时钟同步	无
4	通道切换及通道关闭	C_KS_ADM_SWS	需要时实现通道的切换及通道关闭	通道选择及切换、通道关闭
5	三取二表决	C_KS_ADM_2OO3	对需要输出的应用数据进行三取二表决	无

6	通道重配置	C_KS_ADM_REC	实现正常启动时三个通道的配置过程，以及故障通道修复后重新加入系统的重配置过程	通道启动管理、配置数据获取、通道数据更新

如上表所示，通道间信息管理一级 CSC 包括六个二级 CSC：分别为通道间通信、通道间状态检测、时钟同步、通道切换及通道关闭、三取二表决和通道重配置。其中通道间通信完成各计算通道间的所示数据交换功能，是其它 CSC 完成系统功能的基础，即其它 CSC 需要向伙伴通道发送数据及从伙伴通道接收数据都要经过通道间通信；时钟同步 CSC 比较独立地完成三个通道间的时钟同步，但需要通过通道间通信功能与伙伴通道进行时钟信息交换，当发现时钟有问题时，触发通道切换及关闭 CSC 进行处理；三取二表决 CSC 对应用输出数据进行表决，当表决有问题时，触发通道切换及关闭 CSC 进行处理；通道间状态检测 CSC 维护三个计算通道间的心跳信息，检测通道间的链路问题，当发现问题时触发通道切换及关闭 CSC 进行处理；通道切换及关闭 CSC 主要根据其它 CSC 的触发及计算通道状态信息，进行通道的切换及关闭处理。

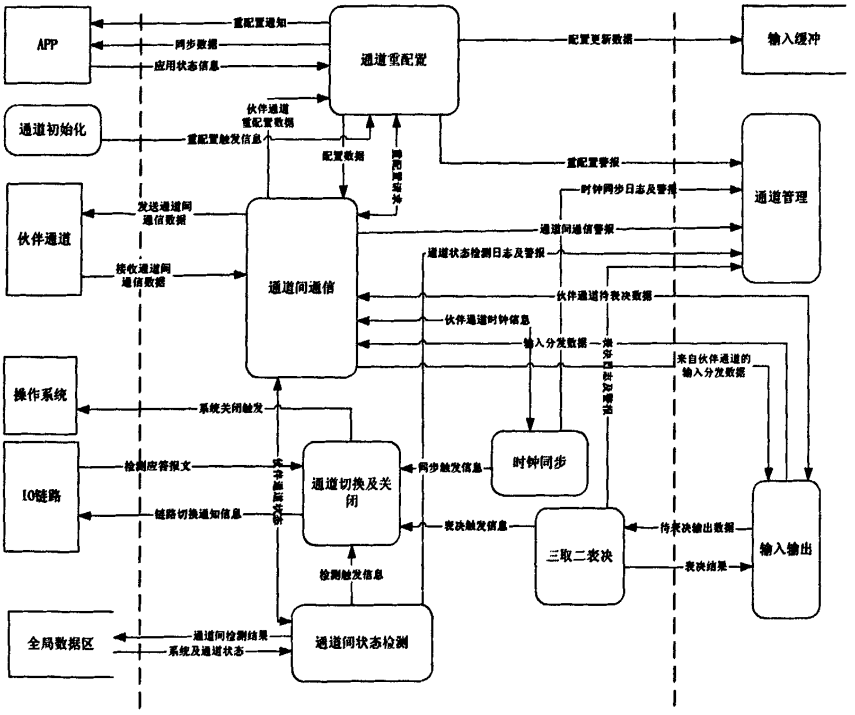

图 KS通道间信息管理二级数据流图

图 5.4.6 通道间信息管理二级 CSC 接口关系

图 5.4.6 通道间信息管理 CSC 包括六个二级 CSC 之间的接口关系进行了说明。

输入输出管理 CSC 包括两个二级 CSC：输入管理和输出管理。这两个 CSC 比较独立分别对应用输入及应用输出进行管理，之间没有直接的数据交换。

表 5.4.4　输入输出管理二级 CSC

序号	二级 CSC 名称	CSC 标识	CSC 用途	下一级 CSC 名称
1	输入管理	C_KS_IO_IN	负责对应用数据的输入及分发	无
2	输出管理	C_KS_IO_OUT	将经三取二表决后的应用数据输出到 IO 板	无

表 5.4.4 对输入输出管理所包含的二个二级 CSC 进行了说明。

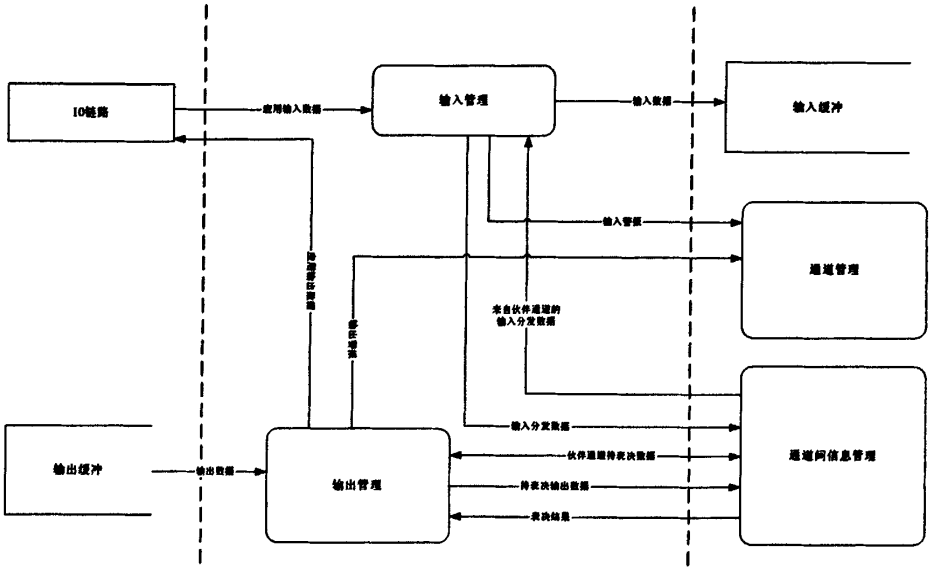

图5.4.7　输入输出管理二级CSC接口关系

图 5.4.7 对输入输出管理二级 CSC 之间的接口关系进行了说明。

自诊无下级 CSC。

5.4.4 CPU 模块主程序设计

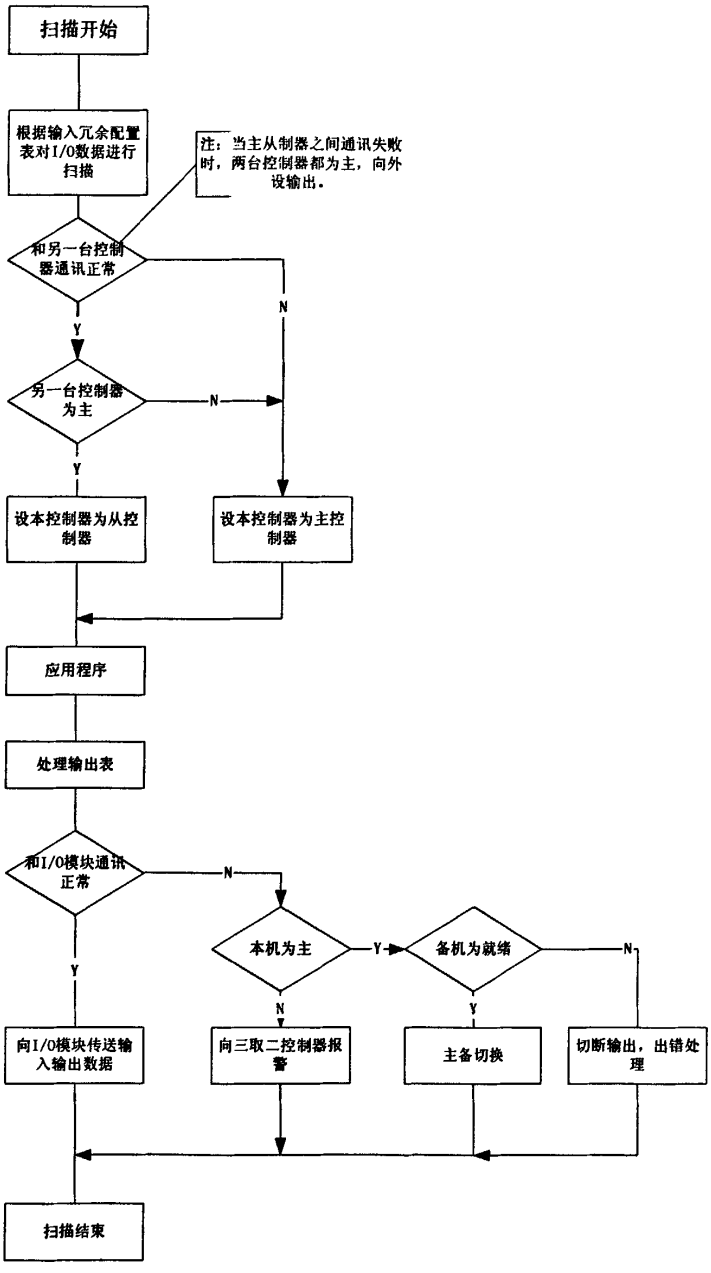

图 5.4.8 CPU 板主程序框图

图 5.4.8 为 CPU 板的主程充流程图，主要说明上电后主备状态的定义和切换过程。

5.4.5 程序示例

```
/*****************************************************************************
*
```

```
* _teachProcess - 教育处理过程
* 处理伙伴通道加入时的教育过程，将自身内存中的相关数据发送到待加入的计算通道
*
*******************************************************************************/
void _teachProcess()
{
    /*测试代码，产生一组测试数据，用以测试学习过程*/
    /*int i,j,datasize;
    unsigned char datatemp[ELEMENTLEN];

    _set_expect_dataID(0,1);

    for (j=0;j<3;j++)          /*产生一组测试程序*/
    {
        datasize=370;
        datatemp[0]= j;
        for(i = 1; i < datasize; i++)
        {
            datatemp[i] =i;
        }
        buffer_write(iTeachBufferID,15,datasize,datatemp);
    }*/

    _set_expect_dataID(iTeachBufferID,1);
    /*调用应用系统提供的教育函数*/

    printf("prepare to teach add xxy");
    teach(iTeachBufferID); /*调用教育函数*/

    _commandSend(iTeachBufferID,22);
}

/*******************************************************************************
*
* _teachProcess - 学习处理过程
* 处理伙伴通道加入时的学习过程，收到伙伴通道送来的数据，并更新自身内存中的相关数据
*
*******************************************************************************/
void _studyProcess(
    int datalength,
    unsigned char * data
    )
{
    printf("Study data process.....\n");
```

```
    learn(datalength,data);        /*调用学习函数*/
    learnTSA(datalength,data);
}
```

示例程序为两个子程序，教育程序和学习程序的测试程序，因为三取二机器在一个机器上电后，对于掉电机器来说，需要学习其余二台机器的状态和中间数据，而对于另外两台正常运行的机器，则存在对第三台需要加入机器的教育过程。运行结果表示为系统状态由二取二升级为三取二状态。

6 结论

通过对地铁用智能 I/O 控制器的设计和实现，我的总结如下 ：

基于 ARM 的安全智能 I/O 控制器具采用通用 ARM7 作为主处理器，系统设计时遵循模块化、通用性的原则，采用无单点硬件故障的冗余设计，本通用智能 I/O 控制器具有如下特点：

1) 采用冗余设计，即任何时候都有双套硬件在同时运行，可根据用户需要组织硬件结构，单个硬件的故障不影响 I/O 控制器的运行。采用模块化设计，能够通过简单地增加或者删减模块数量，调整整体系统处理能力[39]；
2) 具有通用性，内部模块以及外部设备之间，采用标准的接口和传输协议保证良好的可扩展性，很容易和外部设备接口；
3) 采用了嵌入式操作系统，加强了系统的安全性和实时可靠性；
4) 在系统的设计中，按照欧洲信号标准贯穿产品开发的整个生命周期；

通过这次开发，我有以下体会：

1) 从产品开发的角度而言，地铁用安全型智能 I/O 的功能和其它大型系统相比并不复杂，甚至可以说用一台单片机就可以实现所有功能，但是为了提高系统的可靠性、可维护性、可用性和安全性，在硬件、软件方面采用了很多技术，构造成了一个系统。
2) 对于安全相关的产品开发，并不需要硬件、软件越复杂越好[40]。就拿软件而言，并不需要复杂的数据结构和算法，而是要尽量的简单清晰，例如对于指针和递归是限制使用的，对于一些目前大家都认可的高级语言如 C++在安全相关的软件中是被欧洲标准和国内专家强烈推荐不要使用的。
3) 从软件方面提高系统的安全、可靠性大部分是以时间、空间为代价的，完全依靠软件是不行的，如要在一个扫描周期内进行全面深层次的自检是不现实的，只有根据自检的重要性，合理安排，有些工作是一个周期内必须完成的，如对输入、输出口的自检，有些工作可以分散到几个周期内完成，如对寄存器的自检，同时软件应和硬件结合，仅靠软件或硬件是不够的。

在开发的过程中，我遇到了很多问题，主要是如何围绕提高系统的安全可靠性和效率方面的，我在文中重点介绍的解决问题的软件方面的方法，不但可以在计算机联锁输入输出中使用，而且对于其它软件也是可行的。

当然我们的这套系统仍存在一些需要改进的地方，比如目前的系统虽然安全、可靠，但是成本较高，同时所有世界知名厂家都有自己完整独立的信号设备，对外不开入，这就造成安全型输入输出设备成为通用化产品的难度。

我们在不断完善现有输入输出控制器的同时，也对现有的输入输出控制器进行技术上的总结，准备下一代的输入输出控制器的设计方案，计划改进现有的硬件平台，准备采用 VME 总线来组建拥有自主知识产权的安全计算机件平台，从而把输入输出控制器做成通用的安全计算机平台，采用嵌入式操作系统 Vxworks，编程语言采用结构化的 C 语言。软件方面也将更完善，采用更多可以模拟测试的方法来提高安全可靠性。

致谢

在论文完成之际，我向尊敬的指导教师张重雄教授表达深深的敬意和谢意，感谢他们在我攻读工程硕士期间对我的悉心指导和无私帮助，正是由于他们的大力支持，我的工程硕士论文才能够顺利完成。两位导师严谨的治学态度、渊博的知识、开阔的视野和诲人不倦的精神给我留下了深刻的印象，使我受益匪浅、终生难忘。同时，使我观察问题、思考问题、解决问题的能力得到了很大提高。

在论文撰写过程中，中国电子科技集团公司第十四研究所殷浩研高工给予我大力的支持和帮助，在此表示衷心的感谢！同时，对所有关心和帮助我的老师、同学们和同事们表示深深的感谢。

最后，感谢我的家人，没有他们的支持，我可能完成不了工程硕士的学业。无论是入学考试，还是论文撰写，他们给了我信心，使我战胜自己，他们是我前进的源源不断的动力。

参考文献

普通图书 M，会议录 C，汇编 G，报纸 N，期刊 J，学位论文 D，报告 R，标准 S，专利 P，数据库 DB，计算机程序 CP，电子公告 EB。

电子文献载体类型标志如下：磁带 MT，磁盘 DK，光盘 CD，联机网络 OL。

1 杨浩，何世伟. 铁路运输组织学. [M] 第一版. 北京:中国铁道出版社. 2001

1 范成嘉. 浅谈铁路信号新技术的发展趋势. [J] 黑龙江科技信息. 2007（17）：20-23

2 李仕涌，谭南林. 多任务开发系统在嵌入式开发系统中的应用. [J] 北方交通大学学报. 2002（4）：18－21

3 (美)JEAN J.LABROSSE. 嵌入式实时操作系统 μC/OS-II. [M] 第二版. 北京. 北京航空航天大学出版社, 2003

4 陈明计，周立功. 嵌入式实时操作系统 Small RTOS51 原理及应用. [M] 第二版. 北京. 北京航空航天大学出版社, 2005

5 罗蕾. 嵌入式实时操作系统及应用开发(1CD). [M] 第三版. 北京. 北京航空航天大学出版社, 2004

6 张雄伟，陈亮等. DSP 集成开发与应用实例. [M] 第二版. 北京. 北京航空航天大学出版社, 2002

7 程佩青. 数字信号处理. [M] 第四版. 北京. 清华大学出版社, 1995

8 Heinz Kantz, Nikolaus Konig(奥地利). 铁路应用安全计算机平台. [J] 阿尔卡特电信技术展望. 2004（8）：33－48

9 EN50159-1:"Railway applications-Communication, signaling and processing systems, part 1:Safety- related communication in closed transmission systems". [S] 1nd ed. Europen, 1997

10 EN50159-2:"Railway applications-Communication, signaling and processing systems [S] 1nd ed. Europen, 1997

11 张锡弟. 铁路调度工作现状和展望. [M] 第一版. 北京. 中国铁路出版社, 1994

12 闫剑平，步兵. 高速铁路列车定位技术研究. [J] 北方交通大学学报. 1999（6）：7－11

13 Hans-Werner Renz(Canada). 安全、保密的铁路列车控制系统的开放式通信标准. [J] 阿尔卡特电信技术展望. 2004（7）：29－33

14 黄锡滋. 软件可靠性与安全性. [M] 第二版. 北京. 北京清华大学出版社, 1993

15 赵志熙. 从电气联锁到计算机联锁. 北京. [J] 北方交通大学学报 1998（11）：5－9

16 朱得天. 铁路信号基础. [M] 第三版. 北京. 中国铁路出版社, 1987

17 梅登华. 周美玉. 铁路信号的检测和控制系统. [J] 西南交通大学学报. 1996（6）：21－26

18 徐中伟. 安全软件测试理论和技术的研究及其在铁路信号安全软件测评中的实现. [M] 第一版. 北京. 中国铁路出版社, 2000

19 吴汶麒. 城市轨道交通信号与通信系统. [M] 第一版. 北京. 中国铁路出版社, 1998

20 王选. 软件测试方法. [M] 第二版. 北京. 兵器工业出版社. 1992

21 M. Steiner(German). 联锁系统的技术演进战略. [J] 阿尔卡特电信技术展望. 2004（1）：27－30

22 赵志熙. 车站信号控制系统. [M] 第五版. 北京. 铁道出版社, 1994

23 袁由光，陈以农. 容错与避错技术及应用. [M] 第二版. 北京. 北京航空航天大学出版社. 1992

24 赵志熙. 计算机联锁系统技术. [M] 第四版. 北京. 铁道出版社, 1995

25 朱艳军，开祥宝，潘明. 基于CAN总线的I/O模块的设计. [M] 第一版. 北京. 中国铁路, 2003

26 张新民，刘海洋. 二取二制式计算机联锁系统中的通信技术. 第一版. [M] 北京. 中国铁道科学, 2005

27 吴芳美. 铁路安全测试软件评估. [M] 第二版. 北京. 中国铁道出版社, 2001

28 EN 50126—1999 Railway applications - The specification and demonstration of Reliability, Availability, Maintainability and Safety (RAMS). 铁路应用 可靠性，可用性，可维护性和安全性技术条件和验证（RAMS）[S] 1nd ed. Europen, 1999

29 EN 50129 Railway Applications - Safety-related Electronic Railway Control and Protection Systems EN 50129*:铁路应用-安全相关电子控制和保护系统软件. [S] 1nd ed. Europen, 1999

30 周立功. ARM嵌入式系统基础教程. [M] 第二版. 北京. 北京航空航天大学出版社, 2003

31 周立功. ARM嵌入式系统学习指导. [M] 第一版. 北京. 北京航空航天大学出版社, 2003

32 周立功. 常用ARM指令集及汇编. [M] 第一版. 北京. 北京航空航天大学出版社, 2003

33 周立功. LPC2292/2294用户手册（中文）. [M] 第一版. 北京. 北京航空航天大学出版社, 2004

34 邬宽明. CAN总线原理和应用系统设计. [M] 第二版. 北京. 中国科学出版社, 2003

35 李毅力. 二乘（二取二）计算机联锁系统. [J] 计算机工程. 2004（2）：12－16

36 周立功. ARM与嵌入式系统实验教程. [M] 第二版. 北京. 北京航空航天大学出版社. 2003

37 李明. uCOS-II在ARM上的移植. [M] 第二版. 北京. 清华大学出版社, 2006

38 STANDARD EN50128.Railway Applications:Software for Railway Control and Protection Systems. [S] 1nd ed. Europen, 1994

39 张琦. 智能型自律分散调度集中系统研究. [J] 铁道通信信号. 2003 （4）：13－14

40 李国斌. 铁路建设中信号设计方案的选择及系统集成的考虑. [J] 铁道科学研究院通信信号研究所. 2007 （2）：7－9

附 录

A.1 CPU 板的 PCB 图

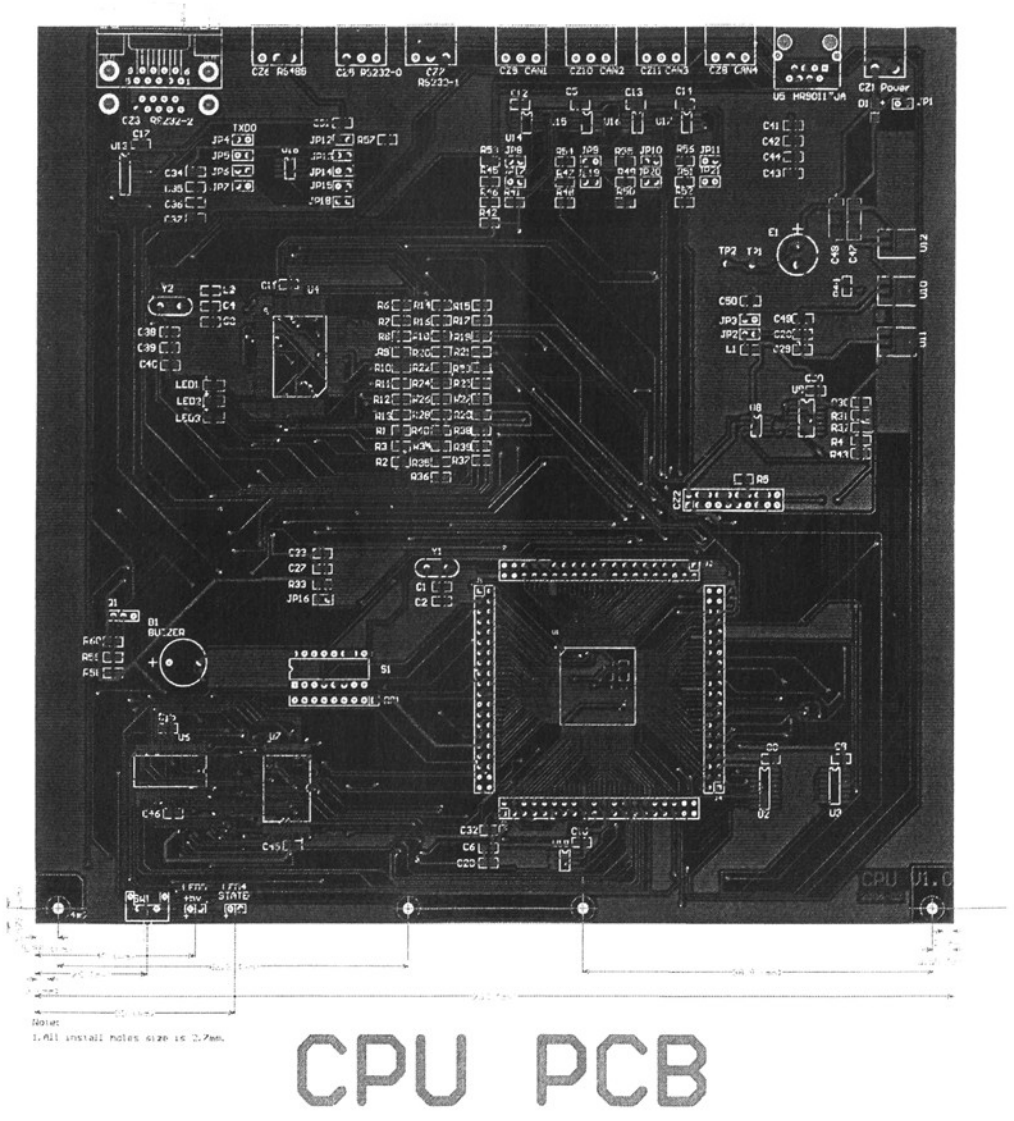

图 A.1 CPU 板的 PCB 图

图 A.1 为 CPU 板的 PCB 图，最上部为外部接口，共有 2 个 232,1 个 485，四路 CAN 和一个以太网接口。右下部为主芯片 LPC2294，左下部为复位开关，热备状态指示和板卡状态指示电路。

A.2 输入板的 PCB 图

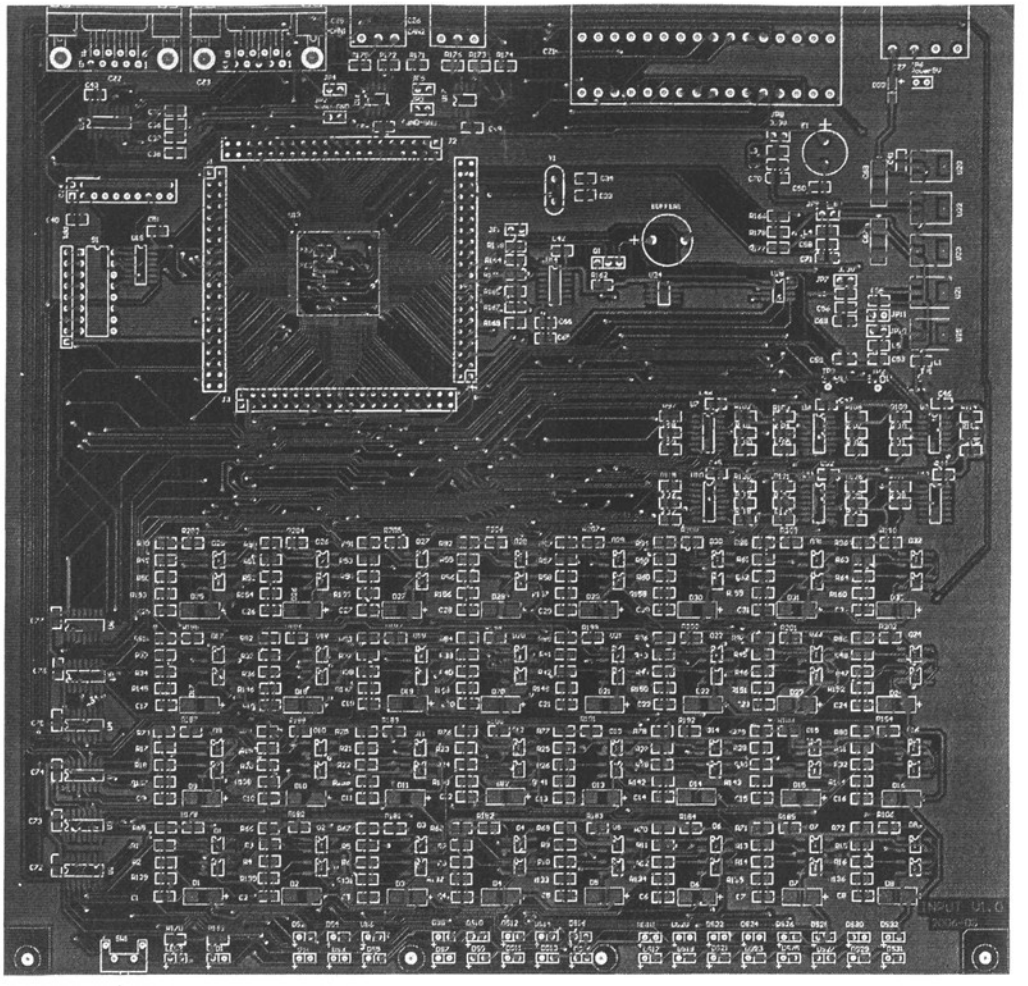

图 A.2 输入板 PCB 图

图 A.2 为输入板的 PCB 图，共有 32 路，最上面为通信接口部份，中间为 CPU 和采集部份，右上部为电源部份，下面为采集和显示部份。

A.3 输出板的 PCB 图

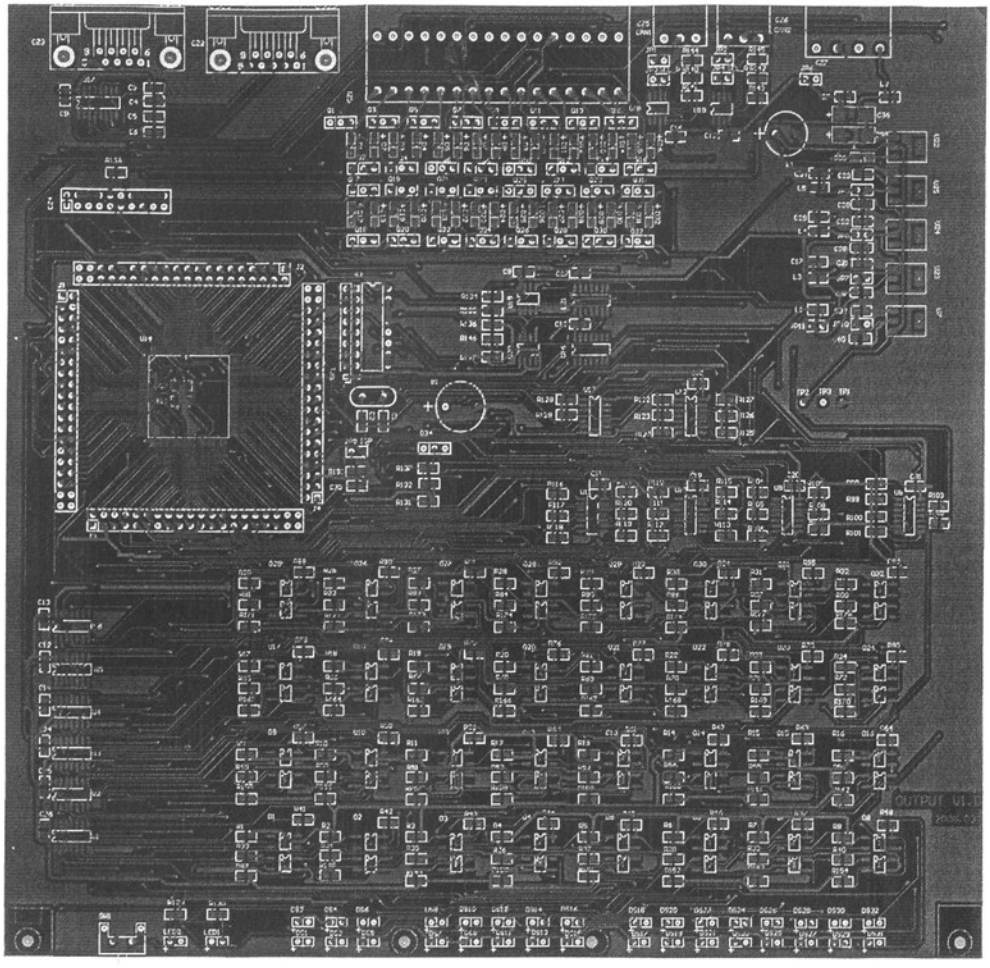

图 A.3 输出板 PCB 图

图 A.3 为输出板的 PCB 图，，它和输入板类似，除电源和输出驱动有差别以外，其余部份是一样的。

A.4 继电器板的 PCB 图

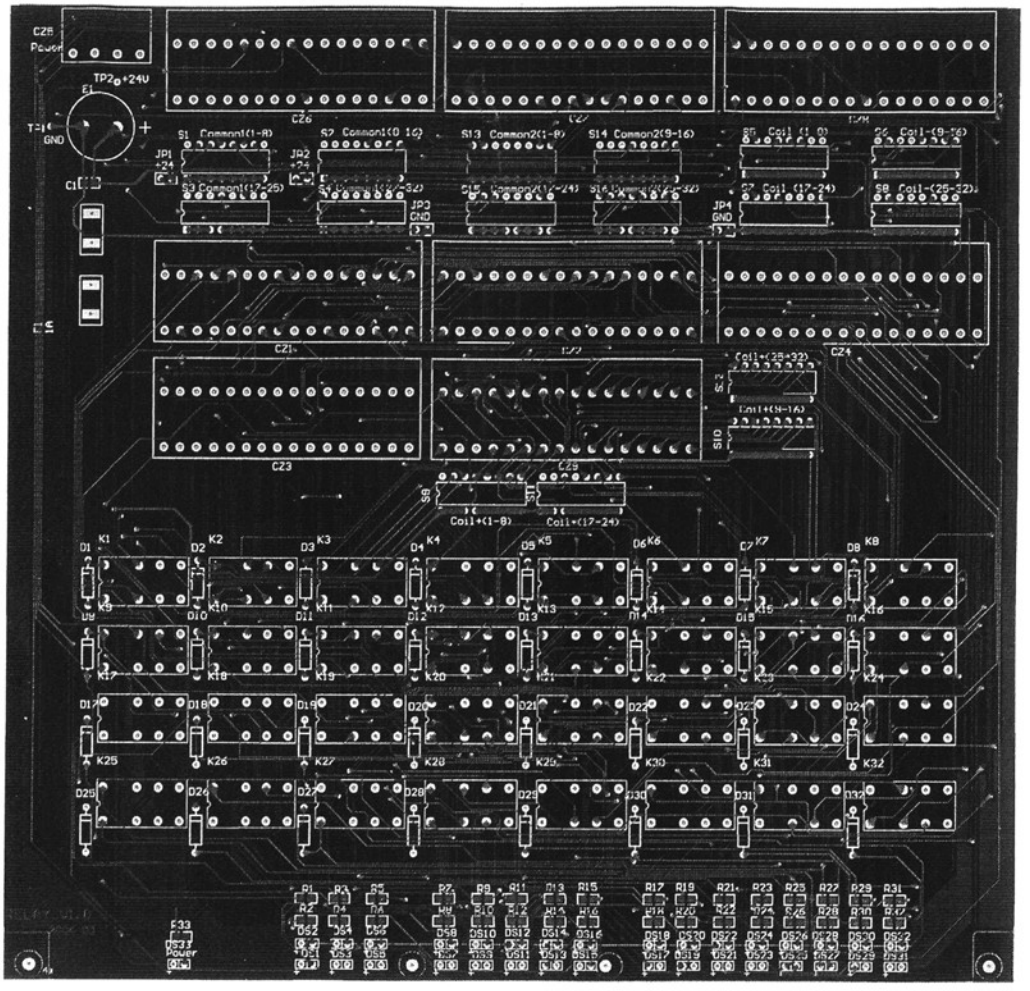

图 A.4 继电器板 PCB 图

图 A.4 为继电器板的 PCB 图，上半部份为插座和地址开关通过开关选择输入接点时，可以选择任一个继电器的任一个接点，也可通过开关选择继电器的线圈的驱动方式 ，如一端接地，一端驱动，或 2 端驱动。