

基于 CAN 总线的测控系统设计

基于 CAN 总线的分布式数据采集及控制系统，主要组成包括 IPC、CAN-bus 接口卡、CAN 功能模块等单元。对于通用 CAN 功能模块可以选择市场产品进行配置，也可以自行开发各类功能模块。如何建立一个现场总线的 CAN 网络，并保证其能够长期稳定地运行，涉及系统硬件、软件以及 CAN 协议正确设计与应用，需要对网络传输数据的流量、设备响应时间等参数进行正确的配置。

8.1 CAN 网络测控系统设计方法

8.1.1 项目需求的分析

采用 CAN 网络进行工程设计，首先要分析系统的需求。例如，现场所需要 I/O 的数量以及 I/O 的类型，决定采用 CAN 节点或者 CAN 功能模块数量，在此基础上进行初步的系统设计，包括如何实现 CAN-bus 网络与测控模块的连接，采用什么样的 CAN-bus 接口卡与工控机连接。项目需求分析是个反复的过程，应该完整全面地考虑，否则会给后续的工作带来较大的影响。在最开始的网络规划中，应注意获取以下的信息：

- 根据系统的设备情况，推算出整个系统的控制规模，决定系统所需要的模块种类和数量，制定 CAN 功能模块清单；
- 获取各个模块对于通讯方面的性能指标要求，包括对实时性、确定性、可重复性的要求，通讯数据量的大小以及 I/O 数据输入输出运行的最大时间间隔等；
- 了解节点或模块对网络通讯的需求，哪些模块涉及的数据对时间有苛刻的要求，哪些模块涉及的数据对时间没有特殊的要求；
- 确定现场信号测量和控制所需要的准确性，模拟量转换的分辨率和精确度，系统采样的最小周期；
- 了解系统中设备的分布情况，包括设备的位置，设备之间的距离，决定系统的网络拓扑、布线、安装方式等，了解系统电源的配置情况；
- 网络工作的环境条件，例如温度、湿度、振动、地磁干扰等；
- 根据系统的要求和设备情况，制定系统相应的控制功能，考虑系统故障几率以及故障情况下系统控制的安全性；
- 系统软件的选择以及配置：开发环境，组态软件，OPC-Server 接口等，系统应用程序的大小，对实时性的要求及影响。

项目需求的分析是整个 CAN 系统设计的基础，必须全面考虑各种因素，避免系统的功能出现遗漏或者欠缺，应把可靠性的要求贯穿到整个系统设计过程中，保证系统最终在现场

中能够可靠稳定的运行，并方便用户根据需要增加其应用功能。

8.1.2 CAN 网络的规划

CAN 系统为基于 PC 机的分布式数据采集及控制系统，基本组成单元为：IPC+CAN-bus 接口卡+CAN 功能模块。CAN 网络在应用中，需要根据实际需求选择合适的控制平台和 CAN 功能模块。对于网络的拓扑结构需要，根据实际布线安装要求、网络通讯速率和通讯距离进行确定，并根据需要增加 CAN-bus 网关/网桥设备，如图 8-1 所示。

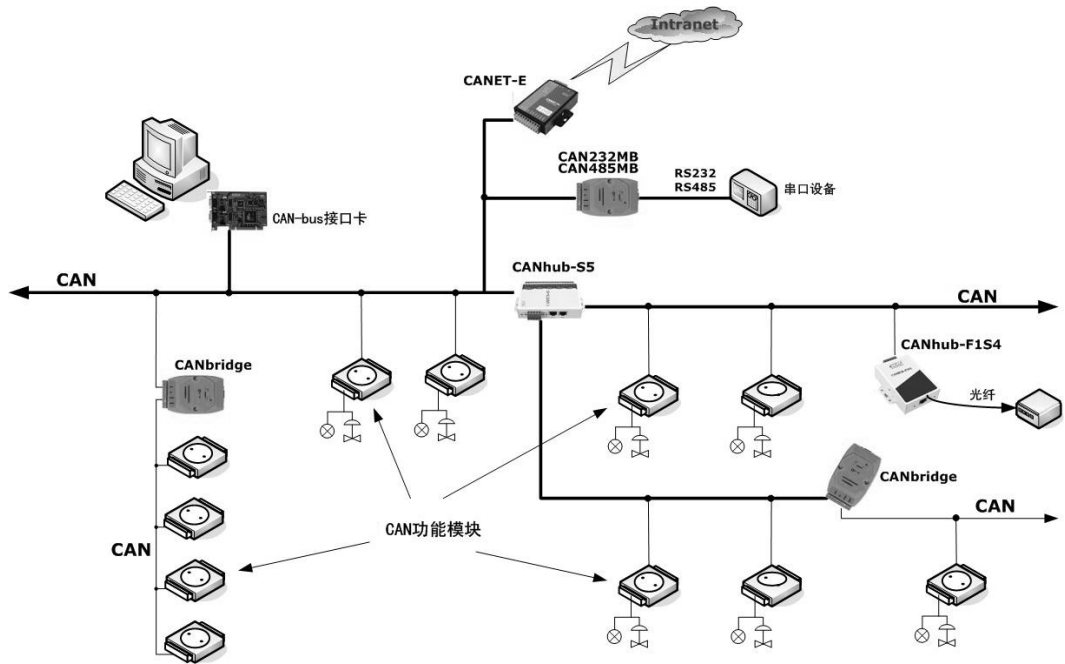


图 8-1 CAN 系统的网络结构

1、控制平台的选择

控制平台是 CAN 系统中的核心单元，是主控设备或主机。控制平台实现了对于整个系统运行的控制以及状态的监控。在 CAN 系统控制平台的选择上，可以采用通用 IPC、嵌入式设备等。

2、功能模块的选择

在工业现场常见的信号类型包括：数字量信号、模拟量信号、脉冲信号以及温度信号。对于现场信号的采集通过 CAN 功能模块实现，CAN 功能模块主要包括 AO、AI、DI、DO、PI 以及一些混合信号类型的 I/O 功能模块。

3、网络拓扑结构

CAN-bus 网络均为总线式拓扑，在工程应用中，由于工业控制现场的环境、设备的分布以及地理位置的要求，总线型拓扑结构往往不能够满足实际布线和安装的要求。可以在 CAN 系统中增加 CAN 中继器、网桥设备，实现对于 CAN 系统网络结构的拓展，如图 8-2 所示。

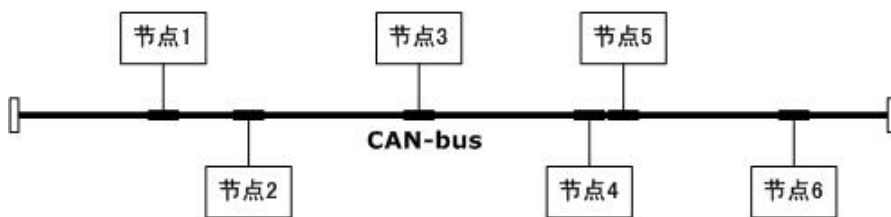


图 8-2 总线式拓扑

4、CAN 网络的性能

CAN 网络系统设计需考虑多种性能，其中系统的实时性影响的因素主要有两种：网络的延时、总线的通讯速率。因此在构建网络时必须确定两个参数。当总线的通信速率较快，则报文传输的时间相对较短，而较高的通信速率会导致较短的传输距离，因此确定这两个参数必须考虑整个 CAN 网络的范围。

8.1.3 CAN 网络的测试

网络的测试是整个系统进入现场安装或者投入实际运行必须进行的环节，是整个系统的基本设计、性能的检测保证。测试主要包括检测主站设备与从站设备之间的通信协议、数据传输的延时以及网络的实时性是否符合设计的要求。图 8-3 所示为网络测试框图，整个系统网络通信、I/O 功能以及运行策略都需要进行详细的测试。

测试过程中要按照真实的网络参数构建整个系统，按照实际的网络拓扑、实际的设备数量以及实际的通信距离连接各个设备，并按照实际的通信参数（总线负载、轮询周期、网络协议以及通信协议）进行测试。在网络通信测试时，底层的 I/O 功能模块可以不连接实际的现场设备。通过对测试结果进行统计分析，评估是否符合预期的设计要求。如果不满足要求，则需要对网络的参数进行修改，直至达到设计要求为止。

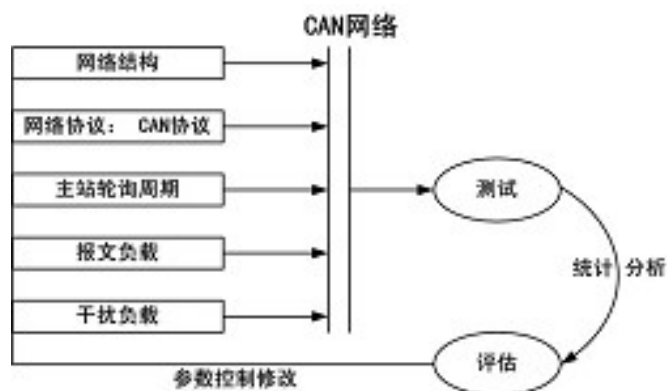


图 8-3 网络测试框图

8.1.4 系统可靠运行的策略

CAN 系统的可靠性依赖于整个系统的硬件和软件的可靠性，在 CAN 系统的设计和测试

中必须对系统的可靠性予以足够的重视。提高 CAN 系统的硬件可靠性可以从以下几个方面：

- 环境适应性方面，在应用时，需要考虑的参数主要有温度、湿度、振动、防水、防尘等；
- 抗电磁干扰措施方面，为提高 CAN 系统的抗干扰能力可以采取屏蔽、接地以及隔离等措施。

8.2 CAN 总线测控系统常用芯片

8.2.1 CAN 芯片简介

目前，许多大的半导体厂商都生产有各类 CAN 总线控制器、驱动器芯片，其中由 Philips 生产的 CAN 控制器、驱动器占领大部分 CAN 应用领域。Philips 生产的 CAN 总线芯片分为以下几类，如表 8-1 所示。

表 8-1 CAN 芯片一览表

类别	型号	备注
32 位 ARM7 芯片	LPC2119/2192/2194	集成 2 路 CAN 通道，LQFP64 小型封装
	LPC2290/2292/2294	集成 2/4 路 CAN 通道，TQFP64 小型封装
8 位 8xC51 芯片	P87C591	集成 PeliCAN 控制器的增强型 MCU
独立 CAN 控制器	SJA1000	应用最广的独立 CAN 控制器
CAN 驱动器	PCA82C250	通用，远程通讯
	PCA82C251	适合工业控制
高速 CAN 驱动器	TJA1050	适合 DeviceNET 网络、控制应用
	TJA1040	适合汽车电子、DeviceNET 网络
标准 LIN 收发器	TJA1020	适合汽车电子等
其他器件	十多个型号的 DC/DC 模块、其他 CAN 总线元件等	

CAN 总线控制器分为两类，一类为独立 CAN 控制器，另一类是集成于微处理器中的 CAN 控制器。独立 CAN 控制器的优点是可与多种微处理器连接，使用灵活方便，像表 8-1 介绍的由 Philips 公司生产的 SJA1000 就属于这一类。而集成 CAN 控制器的 MCU 在系统设计中有利于减少制版面积、降低系统功耗、增强系统抗干扰能力等。无论是哪家产品，都严格遵循 CAN 的规范和国际标准。

CAN 总线驱动器根据是否具有隔离功能分为两类，一类是无隔离 CAN 驱动器，只能实现总线发送功能和接收功能，在工业系统应用中必须增加隔离功能来提高系统的整体抗干扰能力，使用最广泛的是 Philips 公司的 PCA82C250；另一类是带隔离功能的驱动器，集成了高速电气隔离、电源隔离、CAN 收发器，总线保护器于一体，可以实现减少 PCB 制版面积，方便系统设计，提高抗干扰能力的功能，例如广东致远电子有限公司生产的 CTM 系列 CAN 收发器模块就是这一类的典型芯片。

1、独立 CAN 控制器 SJA1000

SJA1000 是 CAN 总线系统设计中最常用的 CAN 控制器，其成本较低，产品和相关技术资料较为成熟和丰富。

(1) SJA1000 的特点

SJA1000 是一款独立的控制器，它是 Philips 公司的 PCA82C200 CAN 控制器的替代产品，具有完成 CAN 通信协议所要求的全部特性。经过简单总线连接的 SJA1000 可完成 CAN 总线的物理层和数据链路层的所有功能。其硬件与软件设计和 PCA82C200 的基本 CAN 模式（BasicCAN）兼容。同时，新增加的增强 CAN 模式（PeliCAN）还可支持 CAN2.0B 协议。SJA1000 的主要特性如下：

- 管脚及电气特性与独立 CAN 总线控制器 PCA82C200 兼容；
- 扩展接收缓冲器（64 字节 FIFO）；
- 支持 CAN2.0B 协议，同时支持 11 位和 29 位标识符；
- 位通讯速率最高可达 1Mbit/s；
- 增强 CAN 模式（PeliCAN）；
- 采用 24MHz 时钟频率；
- 支持多种微处理器接口；
- 可编程的 CAN 输出驱动配置；
- 更宽范围的温度适应能力（-40~125℃）。

(2) SJA1000 的内部功能模块

SJA1000 独立型 CAN 总线控制器功能模块如图 8-5 所示，由以下几部分构成：

- 接口管理逻辑

它接收来自微处理器的命令，控制 CAN 寄存器的地址，并为微处理器提供中断和状态信息。

- 发送缓冲器

13 字节长，位于 CPU 位流处理器（BSP）之间，能存储一条将在 CAN 总线上发送的完整的报文，报文由 CPU 写入，由 BSP 读出。

- 接收缓冲器

（RXB, RXFIFO）：是 CPU 和接收滤波器之间的接口，用来存储从 CAN 总线接收并通过了滤波的报文。接收缓冲器 RXB 是提供给 CPU 可访问的 13 字节的窗口，这个窗口是属于接收 FIFO（RXFIFO）的一部分，共有 64 字节长。有了这个 FIFO，可以在 CPU 处理一个报文的同时继续接收其他到来的报文。

- 验收滤波器

它把报文头中的标识符和验收滤波寄存器中的内容进行比较，以判断该报文是否被接收。如果被接收，报文存入 RXFIFO。

- 位流处理器

它是一个控制发送缓冲器、RXFIFO 并行数据和 CAN 总线（串行数据）之间数据的序列发生器，同时它也执行错误检测、仲裁、位填充和 CAN 总线错误处理功能。

- 位定时逻辑

将 SJA1000 同步于 CAN 总线上的位流。

- 错误管理逻辑

按照 CAN 协议完成错误界定。

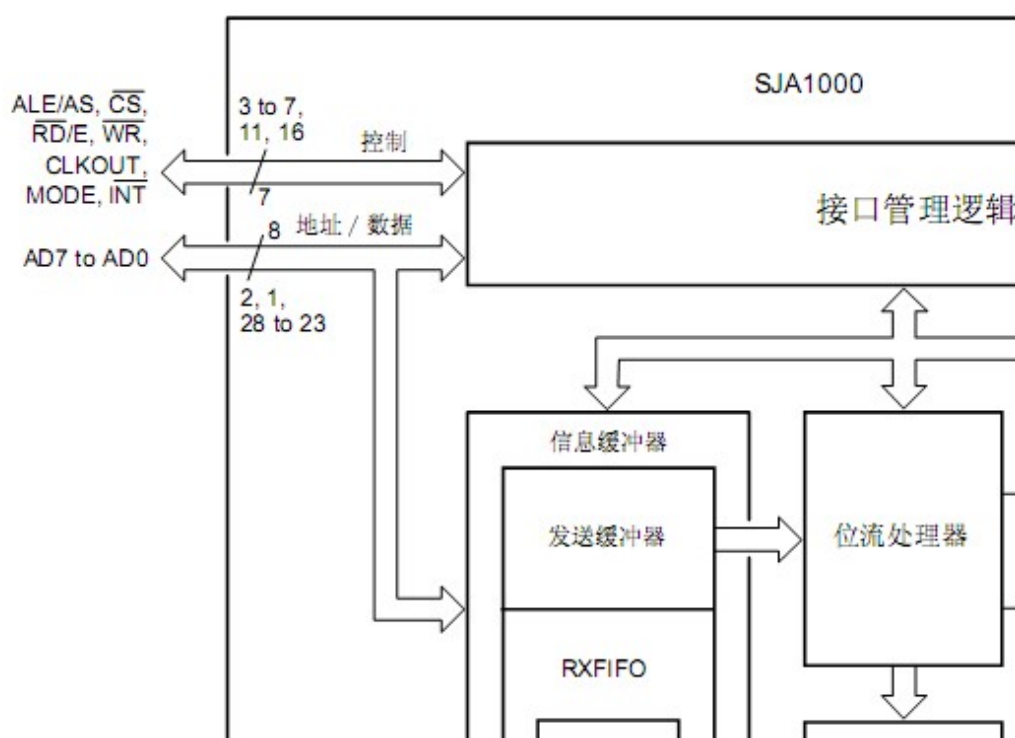


图 8-4 SJA1000 内部结构图

（3）SJA1000 的引脚说明

SJA1000 的引脚可以分为 4 部分：一是电源和地接口，包括信号电源、输入比较器电源和输出驱动器电源；二是数据总线接口，该接口支持 Intel 和 Motorola 两种微处理器；三是振荡器接口；四是 CAN 总线信号接口，包括输入信号 RX0/RX1 接口和输出信号 TX0/TX1 接口。具体引脚功能定义如表 8-2 所列。

表 8-2 SJA1000 引脚功能定义

引脚分类	符号	引脚	功能
电源接口	VDD1	22	逻辑电路的 5V 电源
	VSS1	8	逻辑电路的接地
	VDD2	18	输入比较器的 5V 电源
	VSS2	21	输入比较器的接地端
	VDD3	12	输出驱动的 5V 电压源输入

	VSS3	15	输出驱动器接地
振荡器 接口	XTAL1	9	输入到振荡器放大电路，外部振荡信号由此输入
	XTAL2	10	振荡放大电路输出，使用外部振荡信号时左开路输出
	CLKOUT	7	SJA1000 产生的提供给微控制器的时钟输出信号;时钟信号来源于内部振荡器且通过编程驱动
数据总线 接口	AD7~AD0	2、1、28~23	多路地址/数据总线
	ALE/AS	3	ALE 输入信号(Intel 模式); AS 输入信号(Motorola 模式)
	/CS	4	片选输入，低电平有效
	(/RD)/E	5	/RD(Intel 模式)或 E 使能信号(Motorola 模式)
	/WR	6	/WR(Intel 模式)或 RD/(/WR)信号(Motorola 模式)
	MODE	11	模式选择输入。1=Intel 模式；0=Motorola 模式
	/INT	16	中断输出用于中断微控制器，/INT 在内部中断寄存器各位都被置位时低电平有效
	/RST	17	复位输入，用于复位 CAN 接口（低电平有效）
CAN 总线接口	TX0	13	从 CAN 输出驱动器 0 输出到物理线路上
	TX1	14	从 CAN 输出驱动器 1 输出到物理线路上
	RX0	19	从物理的 CAN 总线输入到 SJA1000 的输入比较器;支配（控制）电平将会唤醒 SJA1000 的睡眠模式;
	RX1	20	

芯片引脚的具体分布如图 8-5 所示。

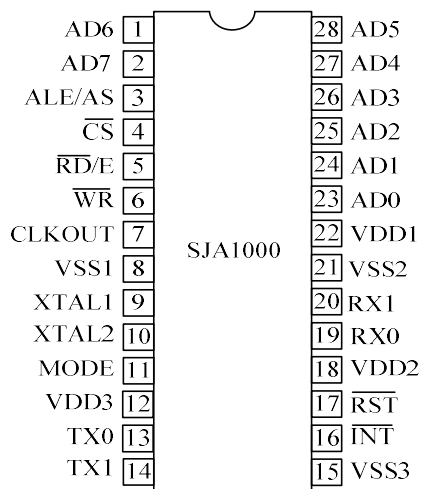


图 8-5 SJA1000 芯片引脚配置图

(4) SJA1000 的设计要点

在利用 SJA1000 构成应用系统时，要注意以下几点设计要点：

在设计微处理器 SJA1000 的接口电路时，首先要根据微处理器合理选择 SJA1000 的接口模式，其次注意 SJA1000 的片选地址应与其他的外部存储器无冲突，还应注意 SJA1000 的复位电路应为低电平有效。

微处理器对 SJA1000 的控制访问是以外部存储器的方式来访问 SJA1000 的内部寄存器，所以应该正确定义微处理器访问 SJA1000 及其内部寄存器的访问地址。

微处理器访问 SJA1000 的方式，可以通过中断或轮询的方式来访问 SJA1000。

微处理器访问 SJA1000 时，有 2 种不同的模式：工作模式和复位模式。对 SJA1000 的初始化只能在 SJA1000 的复位模式下进行。初始化包括（设置验收滤波器、总线定时器、输出设置、时钟分频中的特定控制等）。设置复位请求后一定要校验，以确保设置成功。

向 SJA1000 的发送缓冲区中写入数据时，一定要检查发送缓冲区是否处于锁定状态。如果缓冲区被锁定，这时写入的数据将丢失。

对 SJA1000 的操作难点在于总线定时器的设置。设置总线定时器包括：设置总线波特率、同步跳转宽度、微周期的长度、采样点的位置和在每个采样点的采样数目。计算以上参数的详细公式参见 SJA1000 的数据手册。

2、CAN 总线驱动器 PCA82C250

CAN 总线驱动器就是将控制器发送信号转变为符合 CAN 物理层标准的信号，并进行放大、传输到总线上；并且将总线上接收到的信号转变为控制器所能接收的电平信号。

CAN 总线驱动器根据不同的 CAN 物理层协议分为高速 CAN 总线驱动器和容错 CAN 总线驱动器。注意，高速和容错 CAN 驱动器不能在同一个 CAN 网络中使用。下面要介绍的 PCA82C250 属于高速 CAN 总线驱动器

（1）82C250 收发器的特点

PCA82C250 收发器是 CAN 协议控制器和物理传输线路之间的接口，并对总线提供差动发送能力和对 CAN 控制器提供差动接收能力。其主要特点如下：

- 完全符合“ISO11898”标准；
- 高速率（最高可达 1Mbit/s）；
- 具有抗汽车环境下的瞬间干扰，保护总线能力；
- 采用斜率控制（Slope Control），降低射频干扰（RFI）；
- 差分收发器具有宽范围的抗共模干扰能力，有很强的抗电磁干扰（EMI）的能力；
- 过热保护；
- 总线与电源及地之间的短路保护；
- 低电流待机模式；
- 未上电节点不会干扰总线；
- 总线至少可连接 110 个节点，在汽车环境中，对总线提供瞬变保护；
- 工作温度范围宽：（-40~125）℃。

（2）82C250 收发器引脚介绍

PCA82C250 为 8 引脚芯片，有 DIP 和 SO 两种封装，引脚如图 8-6 所示，引脚介绍如表 8-3 所示。

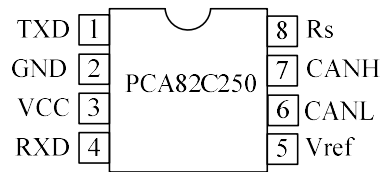


图 8-6 PCA82C250 芯片引脚配置图

表 8-3 PCA82C250 引脚及功能表

符号	引脚号	功能描述	符号	引脚号	描述
TXD	1	发送数据输入	Vref	5	参考电压输出
GND	2	地线	CAN_L	6	输入/输出低电平
VCC	3	电源电压	CAN_H	7	输入/输出高电平
RXD	4	接收数据输出	Rs	8	斜率电阻输入

PCA82C250 驱动电路内部具有限流电路，可防止发送输出级对电源、地或负载短路。虽然短路出现时功耗增加，但不致使输出级损坏。若温度超过大约 160℃，则两个发送器输出端极限电流减小，由于发送器是功耗的主要部分，因而限制了芯片的温升。器件的所有其他部分将继续工作。PCA82C250 采用双线差分驱动，有助于抑制汽车等恶劣电气环境下的瞬间干扰。

引脚 8（Rs）用于选定 PCA82C250 的工作模式。有三种不同的工作模式可供选择：高速、斜率控制和待机，如表 8-4 所列。

表 8-4 引脚 Rs 用法

Rs 提供条件	工作模式	Rs 上的电压或电流
$V_{RS} > 0.75V_{CC}$	待机模式	$-10\mu A < I_{RS} < 10\mu A$
$-10\mu A < I_{RS} < -200\mu A$	斜率控制	$0.3V_{CC} < V_{RS} < 0.6V_{CC}$
$V_{RS} < 0.3V_{CC}$	高速模式	$I_{RS} < -500\mu A$

对于高速工作模式，发送器输出级晶体管被尽可能快地启动和关闭。在这种模式下，不采取任何措施限制上升和下降的斜率。此时，建议采用屏蔽电缆以避免射频干扰问题的出现。通过把引脚 8 接地可选择高速工作模式。

对于较低速度或较短的总线长度，可使用非屏蔽双绞线或平行线作总线。为降低射频干扰，应限制上升和下降的斜率。上升和下降的斜率可以通过由引脚 8 至地连接的电阻进行控制，斜率正比于引脚 8 上的电流输出。

如果引脚 8 接高电平，则电路进入低电平待机模式。在这种模式下，发送器被关闭，接收器转至低电流。如果检测到显性位，RXD 将转至低电平。微控制器应通过引脚 8 将驱动器变为正常工作状态来对这个条件作出响应。由于在待机模式下接收器是慢速的，因此将丢失第一个报文。PCA82C250 真值表如表 8-5 所列。

表 8-5 CAN 驱动器真值表

电源	TXD	CAN_H	CAN_L	总线状况	RXD
(4.5~5.5) V	0	高	低	显性	0

(4.5~5.5) V	1 或悬空	悬空	悬空	隐性	1
<2C (未上电)	×	悬空	悬空	隐性	×
2V<V _{cc} <4.5V	>0.75V _{cc}	悬空	悬空	隐性	×
2V<V _{cc} <4.5V	×	若 $V_{RS}>0.75V_{cc}$ 悬空	若 $V_{RS}>0.75V_{cc}$ 悬空	隐性	×

利用 PCA82C250 还可方便地在 CAN 控制器与驱动器之间建立光电隔离。以实现总线各节点间的电气隔离。

双绞线并不是 CAN 总线的唯一传输介质。利用光电转换按门器件及星形光纤耦合器可建立光纤介质的 CAN 总线通信系统。此时，光纤中有光表示显性位，无光表示隐性位。

利用 CAN 控制器的双相位输出模式，通过设计适当的接口电路，也不难实现人们希望的电源线与 CAN 通信线的复用。另外，CAN 协议中错误抢出及自动重发功能为建立高效的基于电力线载波或无线电介质的 CAN 通信系统提供了方便。

8.2.2 集成有 CAN 总线控制器的微处理器

独立的 CAN 控制器芯片需要外接一个微处理器，接受外部 CPU 的控制才能运行。如果微处理器内带有 CAN 控制器，会大大简化应用系统的硬件设计，系统的可靠性也有很大提高。集成有 CAN 总线控制器的微处理器，不仅与相应的微处理器完全兼容，而且在此基础上增加了许多专用的外围器件。

而随着嵌入式技术的进步，许多 ARM 内核的微处理器也集成了 CAN 控制器，同时拥有大量丰富的外设及其它总线接口，为各种总线的转接及数据融合技术提供了更为丰富的设计方案选择。比如 Philips 公司生产的 LPC2377/2378 微控制器就是基于一个支持实时仿真的 16 位/32 位 ARM7TDMI-S CPU，此外还有像 TI 公司的 LM3S2000/5000/8000 系列 ARM Cortex-M3 处理器内建 1~3 路 CAN 控制器，可同时支持多路 CAN 总线的操作，使器件可用作网关、开关、工业或汽车应用中多个 CAN 总线的路由器。下面以 P8xC591 系列芯片为例进行介绍。

1、P8xC591 的结构功能模块

P8xC591 Philips 是公司近期推出的内带 CAN 控制器的 8 位高性能的微控制器，它采用了强大的 80C51 指令集并成功地包括了 SJA1000 CAN 控制的控制的 peliCAN 功能，图 8-7 为 P8xC591 内部结构方框图。该嵌入式 CAN 控制器包括了下列功能模块：

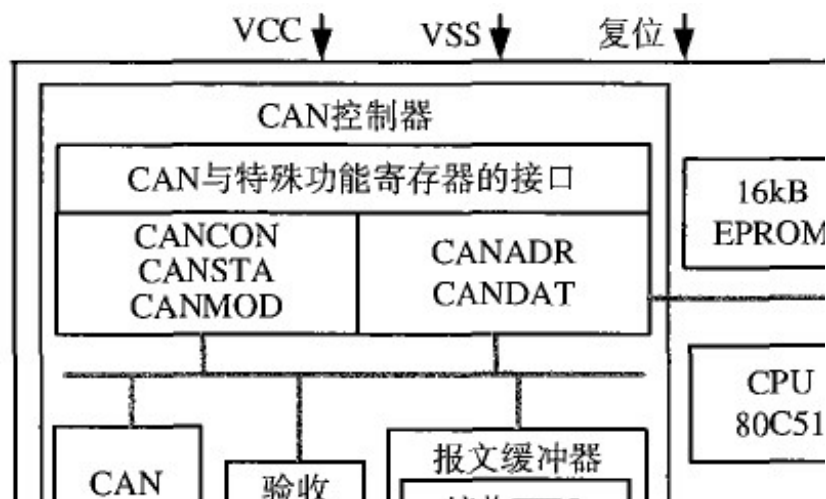


图 8-7 P8xC591 内部结构方框图

- CAN 内核模块：根据 CAN2.0B 规范控制 CAN 帧的发送和接收；
- CAN 接口：包含 5 个实现 CPU 与 CAN 控制器连接的特殊功能寄存器，对重要 CAN 寄存器的访问通过快速自动增加的寻址特性和对特殊功能寄存器的位寻址来实现；
- CAN 信息缓冲区：控制器的发送缓冲区能够保存一个完整的 CAN 信息 扩展或标准帧格式。只要通过 CPU 启动发送，信息字节就从发送缓冲区传输到 CAN 内核模块；当接收一个信息时 CAN 内核模块将串行位流转换成并行数据输入到验收滤波器，通过该可编程滤波器 P8xC591 确定实际接收到的信息。

所有由验收滤波器验收的接收数据都保存在接收 FIFO 中，取决于操作模式和数据长度的不同，最多可保存 21 个 CAN 信息，这使用户在指定系统的中断服务和中断优先级时有更多的灵活性，因为数据溢出的可能性大大降低了。

2、 P8xC591 的相关特性

CAN2.0B 控制器支持 11 位标准和 29 位扩展识别码，使用 8MHz 时钟可实现 1MbpsCAN 总线速率，还具有一个片内 64 字节接收 FIFO 和一个 13 字节发送缓冲区，除了普通的 CAN 特性以外，P8xC591 还提供增强型 PeliCAN 系统的维护、诊断和优化特性。

PeliCAN 特性：

- 4 个独立可配置的验收滤波器组；
- 每个组都有 4 个可选的验收滤波器配置；
- 每个验收滤波器都有 32 位区分符，32 位代码和 32 位屏蔽；
- 所有滤波器都可 在运行中改变；
- 支持更高层的协议的验收滤波器；
- 接收 FIFO 特性；

- 只听模式，自检测模式；
- 只有达到 FIFO 接收中断级才产生接收中断 ；
- 在接收到高优先级数据帧时立即产生接收中断 。

3、 P8xC591 和 SJA1000 CAN 特性比较

P8xC591 组合了 P87C554（微控制器）和 SJA1000 的功能，并具有增强的 CAN 接收中断，扩展的验收滤波器等特性。P8xC591 和 P87C554 之间的主要区别在于片内 CAN 控制器、6 输入 ADC、低电平复位、44 个引脚，与 SJA1000 的比较如表 8-6 所示。

表 8-6 P8xC591 和 SJA1000 CAN 特性比较

SJA1000	P8xC591
1 个支持双或单滤波器的验收滤波器组	4 个支持双或单滤波器的验收滤波器组，所有滤波器都可在运行中改变
1 个接收中断配置：在接收任何信息时都会产生接收中断（接收 FIFO 非空）	2 个接收中断配置：高优先级中断，接收中断级
CAN 位定时计算： $t_{SCL} = 2 / f_{XTAL} \times (32BRP.5 + \dots + 1)$	CAN 位定时计算： $t_{SCL} = 1 / f_{XTAL} \times (32BRP.5 + \dots + 1)$
BasicCAN 模式，PeliCAN 模式	PeliCAN 模式
两个输出管脚（TX0 ，TX1）的不同发送输出级配置：正常，双相位，时钟和测试输出模式	输出管脚（TXDC）的发送输出级配置为正常模式。

4、 P8xC591 收发器引脚介绍

P8xC591 为 44 引脚芯片，有 LCC 和 QFP 两种封装，引脚如图 8-8 所示。

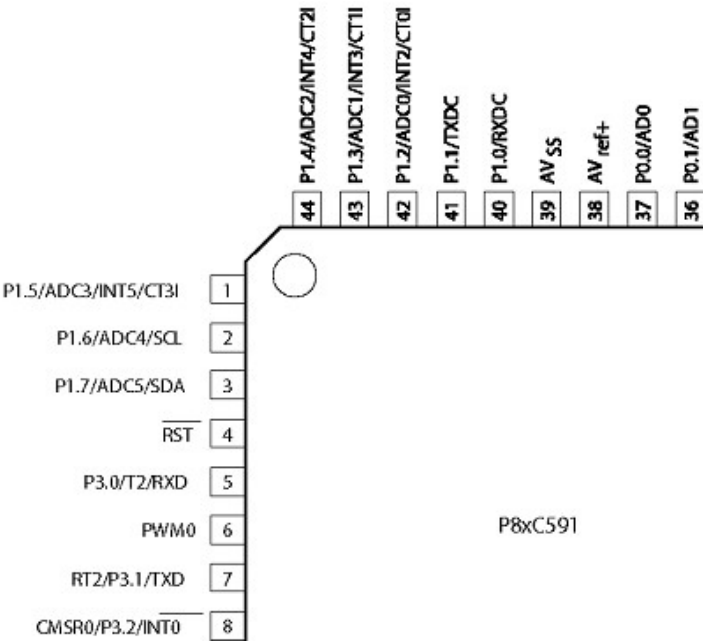


图 8-8 P8xC591 引脚图

8.3 CAN 总线测控系统设计要点分析

CAN 总线接口电路主要包括：单片机、控制器接口、总线收发器等。按照 CAN 总线物理层协议选择总线介质，设计布线方案，连接成 CAN 网络。设计时根据实际需求，选择相应的总体方案，如单片机+独立 CAN 控制器+总线收发器或者带 CAN 控制器的微处理器+CAN 总线收发器方案等。双绞屏蔽线可设两套，在两套介质上同时进行信息传输，接收方只用一个介质。在冗余和非冗余段的连接临界点处进行总线切换。

8.3.1 CAN 总线网络电气连接

CAN 总线网络采用直线拓扑结构，在一个网络上至少存在 2 个 CAN 总线节点。图 8-9 所示为 n 个 CAN 模块通过 CAN 总线通讯网络电气连接示意图。基于 CAN 总线通信的接口电路主要由 CAN 通信控制器与微处理器之间的接口电路以及 CAN 总线收发器与物理总线之间的接口电路构成。CAN 总线控制器主要实现数据链路层规定的功能，提供了与微控制器 MCU 的数据线路接口，CAN 总线收发器实现物理层的功能，提供 CAN 控制器与物理总线之间的接口，为总线提供差分信号的发送功能，为控制器提供差分信号的接收功能。在总线的两个终端，各需要安装 1 个 120Ω 终端电阻。CAN 总线信号采用差分传输方式，所有节点的 CAN_H 连接在一起，CAN_L 连接在一起。设计者只需要根据具体项目要求对 CAN 总线应用层进行设计，并通过对微处理器编程控制 CAN 总线控制器来实现 CAN 网络通信。

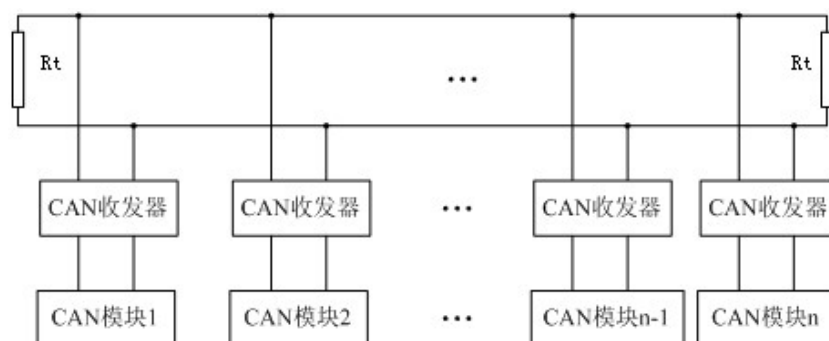


图 8-9 CAN 总线网络电气连接示意图

分布式 CAN 总线数据采集及控制系统中，干线的通讯距离与 CAN 总线网络通讯波特率成反比：在 5Kbps 下能够传输 10km 距离，在 1Mbps 下能够传输 40m 距离。对于支线，其长度一般不应超过 3 米。如果直接采用国际标准规范，则对干线、支线有更严格的要求。

8.3.2 CAN 节点设计

硬件电路的设计主要是 CAN 通信控制器与微处理器之间和 CAN 总线收发器与物理总线之间的接口电路的设计。CAN 通信控制器是 CAN 总线接口电路的核心, 主要完成 CAN 的通信协议, 而 CAN 总线收发器的主要功能是增大通信距离, 提高系统的瞬间抗干扰能力, 保护总线, 降低射频干扰, 实现热防护等。

CAN 节点至少要包括微控制器（MCU）、CAN 协议控制器、CAN 收发器三部分。根据不同的需要可以选择不同的器件。如果有特别的需要在 CAN 收发器与总线之间或者是 CAN 协议控制器与收发器之间加入光电隔离，以提高系统的抗干扰能力。

图 8-10 是 P8xC591 芯片的典型应用图，带有 CAN 控制器的微控制器，都可以采用类似的应用结构，即在控制器和 CAN 总线之间加一个 CAN 总线收发器即可，用于性能要求不高的低端应用场合。

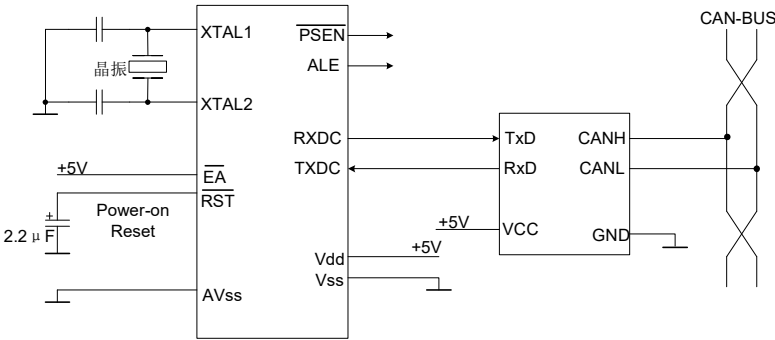


图 8-10 典型的 P87C591 应用图

对于节点性能要求比较高的场合，CAN 总线部分选择独立的 CAN 控制器 SJA1000，微处理器则可以灵活选择各种高速 51、ARM、DSP、PowerPC、X86 等，以适应不同的应用环境。图 8-11 是由 SJA1000 以及 PCA82C250 组成的 CAN 总线最小节点的典型应用电路图。

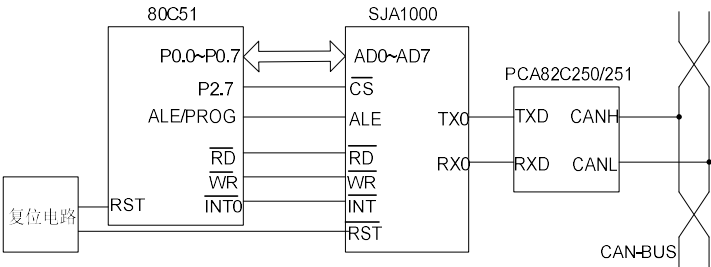


图 8-11 SJA1000 构成 CAN 总线最小节点

图 8-12 给出了一个典型的 CAN-bus 通讯单元电路图，电路结构为 MCU（P89C52）+ CAN 控制器（SJA1000）+ 隔离 CAN 收发器，整个系统电源采用+5V 电源输入，上电复位芯片可保证上电时正确的启动系统。微处理器采用 P89C52 单片机，该系列单片机是 80C51 微控制器的派生器件，采用先进的 CMOS 工艺制造，指令系统与 80C51 完全相同。CAN 控制器采用 SJA1000，该电路中采用了隔离 CAN 收发器模块，以确保在 CAN 总线遭受严重干扰时控制器能够正常运行。

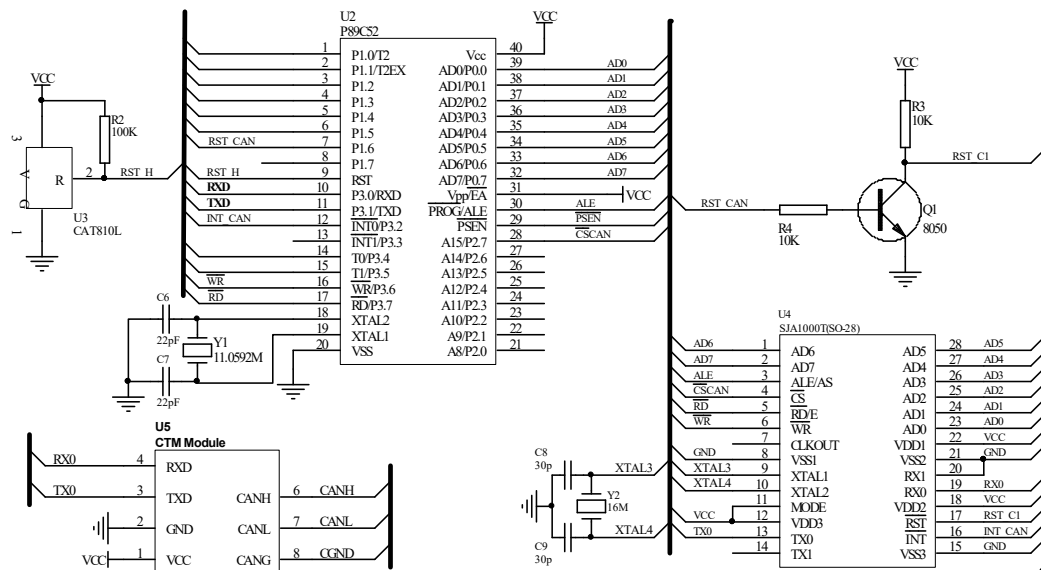


图 8-12 典型的 CAN-bus 通讯单元电路图

8.3.3 CAN 总线系统智能节点软件设计

CAN 总线节点的软件主要包括三大部分：CAN 节点初始化，报文发送和报文接收。熟悉这三部分程序的设计，即可编写 CAN 总线通信的一般应用程序。如果要将 CAN 总线应用于比较复杂的系统中，还需详细地了解有关 CAN 总线错误处理、总线关闭处理、接收滤波处理、波特率参数设置和自动检测以及 CAN 总线通信距离和节点数的计算等方面的内容。

1、 初始化过程

SJA1000 的初始化只有在复位模式下才可以进行，初始化主要包括工作方式的设置、接收滤波方式的设置、接收屏蔽寄存器和接收代码寄存器的设置、波特率参数设置和中断允许寄存器的设置等。在完成 SJA1000 对初始化设置以后，SJA1000 就可以回到工作状态，进行正常的通信任务。下面是在 PeliCAN 函数库基础上提供的用户初始化可操作例程。

```
SJA1000_Config_Normal()
{
    BTR0=0x00;

    BTR1=0x14;           //设置为 1M 波特率通信

    SJAEntryResetMode(); //进入复位模式

    WriteSJAREg(REG_CAN_CDR,0xc8); //配置时钟分频寄存器，选择 PeliCAN 模
式

    WriteSJAREg(REG_CAN_MOD,0x05); //配置模式寄存器，选择双滤波、自发自
收模式

    WriteSJAREgBlock(16,Send_CAN_Filter,8); //配置验收代码/屏蔽寄存器
```

```

WriteSJAReg(REG_CAN_BTR0,BTR0); //配置总线定时器 0x00

WriteSJAReg(REG_CAN_BTR1,BTR1); //配置总线定时器 0x14

WriteSJAReg(REG_CAN_OCR,0x1a); //配置输出管脚，推挽输出。

SJAQuitResetMode(); //退出复位模式，进入工作模式

}

```

2、报文发送过程

发送子程序负责节点报文的发送。发送时用户只需将待发送的数据按特定格式组合成一帧报文，送入 SJA1000 发送缓冲区中，然后启动 SJA1000 发送即可。在往 SJA1000 发送缓冲区送报文之前，必须先做一些判断。发送程序分发送远程帧和数据帧两种，远程帧并无数据域。

以下为一个发送报文的简单编程实例。

```

main()
{
    SJA_CS_Point=&CAN_SJA_BaseAdr;

    SJA1000HardwareRst(); //SJA1000 硬件复位

    SJA1000_Config_Normal(); //SJA1000 进入正常模式配置

    WriteSJAReg(REG_CAN_IER,0x02); //使能 SJA1000 发送中断位

    WriteSJARegBlock(16,Send_CAN_Info_ID,5); //扩展帧，向发送缓冲区写入 5 个数据

    WriteSJARegBlock(21,Send_CAN_Data,8); //扩展帧，向发送缓冲区写入 8 个数据

    while(1)
    {
        canstatus = ReadSJAReg(REG_CAN_SR);

        if((canstatus&0x0c)==0x0c) //判断是否可以发送
        {
            WriteSJAReg(REG_CAN_CMR,1); //使能发送请求，发送数据。

        }
    }
}

```

3、报文接收过程

接收子程序负责节点报文的接收以及其他情况处理。接收子程序比发送子程序要复杂一点，因为在处理接收报文的过程中，同时要对诸如总线关闭、错误报警、接收溢出等情况进

行处理。SJA1000 报文的接收主要有两种方式：中断接收方式和查询接收方式。如果对通信的实时性要求不是很强，建议采用查询接收方式，两种接收方式的编程思路基本相同。

8.4 基于 CAN 总线的测控系统设计

8.4.1 CAN 总线测控系统介绍

本节内容以 CAN 总线实验系统组成框图图 8-13 为基于 CAN 总线构建的通用测控实验平台 CAN 总线系统组成框图。系统体现了网络化以及总线技术的特点。系统通过应用软件来实现各项实验的内容，利用应用软件发出控制指令及相关的数

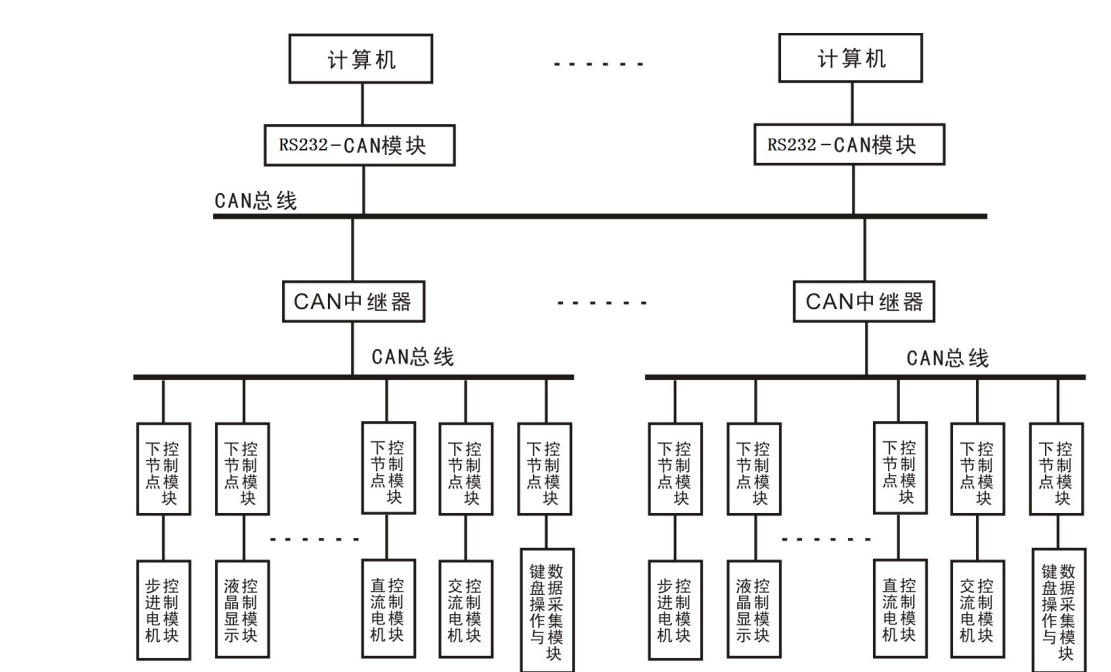


图 8-13 CAN 总线系统组成框图

该系统主要实验内容：

- 标准模拟信号的采集与处理
 - 数字信号的采集与处理
 - 脉冲信号的采集与处理
- 系统集成了标准模拟信号的采集电路，在计算机上可以实时显示采集的数据，并进行数据的处理。
- 系统集成了数字信号的采集电路，在计算机上可以实时显示采集的数字信号，并进行数据的处理。
- 系统集成了脉冲信号的采集电路，在计算机上可以实时显示采集的脉冲信号，并进行

数据的处理。

- 步进电机的控制

通过在计算机上设定步进电机的转角、转向，实现步进电机的控制。

- 直流电机的控制

通过在计算机上设定直流电机的转速、转向，实现直流电机的控制。

- 交流电机的控制

通过在计算机上设定交流电机的转向，实现交流电机的转向控制。

8.4.2 CAN 总线测控系统组成

1、上位机

上位机设计主要是编程实现运行于计算机上的一个应用软件，通过该系统直接地对实际的控制对象进行控制操作。

该应用软件用 VC++ 编程来实现，采用了 SDK 的编程方式，而不是利用 MFC 来实现，该方式编程灵活、可扩充性好，程序占用的存储空间少，但是编程效率低。本系统主要是调用了 API 库函数来打开串口、对串口进行设置与初始化，进而实现上位机与下位机之间的通信。

2、RS232-CAN 转换卡

RS232-CAN 转换卡也叫做 RS232-CAN 网桥，是 CAN 总线测控实验系统中的核心部分。PC 机不能直接对 CAN 总线网络进行监控和跟踪，需要一个 RS232-CAN 网桥配合相应的上位机软件，对 CAN 节点通信状态进行监控。CAN 网络和 RS232 网络的连接，是本节介绍的重点，涉及硬件结构、通信协议设计及程序流程。

3、中继器

中继器主要完成节点扩展与线路延长的功能，实现在两个 CAN 网段之间的数据转发功能。中继器是连接网络线路的一种装置，常用于网络节点之间物理信号的双向转发工作，负责在两个节点的物理层上按位传递信息，完成信号的复制、调制和放大功能以此来延长网络的长度，使用中继器解决了在线路上传输信号衰弱的问题，有效的增加了网络通信的长度。

4、节点控制模块

模块主要完成 CAN 总线通信以及对控制模块发出指令的功能。模块的设计采取了多种接口电路集于一体的设计方案，一个控制模块可以控制多种对象。块集成的电路接口有：步进电机接口、直流电机接口、交流电机接口、液晶接口、按键接口、数据采集接口。CAN 通信电路由 AT89C52、SJA1000、MCP2551 及附加元器件构成，MCP2551 是一个可容错的高速 CAN 器件，可作为 CAN 协议控制器和物理总线接口。模块的设计可参考 8.3 节方法进行。

8.4.3 RS232-CAN 转换卡设计

RS232-CAN 转换卡设计时应满足以下特性：

- 实现 CAN 网络和 RS232 网络的连接，硬件上实现物理层的连接，链路层协议上实现数据链路层的连接；
- 具有一定的存储功能，能够对不同网络的通信报文进行缓存；
- 具有报文滤波功能，检查传输中的报文是否需要转发到另一个网络。

1、CAN-RS232 转换卡硬件结构

CAN-RS232 转换卡在硬件上要具有 CAN 总线接口，具有 RS232 接口，具有一定的存储功能，对 CAN-RS232 转换卡的工作状态具有一定的监控作用。

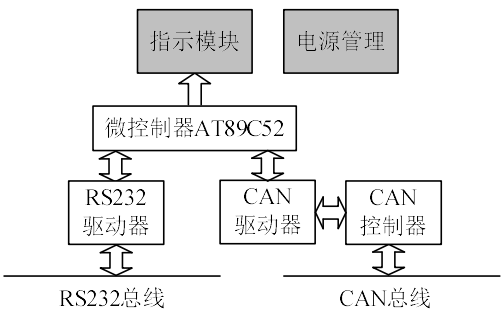


图 8-14 CAN-RS232 转换卡硬件结构图

转换卡的微控制器采用 AT89C52，负责网桥的监控任务。AT89C52 是一个低电压，高性能 8 位单片机，片内含 8k bytes 的可反复擦写的 Flash 只读程序存储器和 256 Bytes 的随机存取数据存储器，用于报文的缓存；CAN 总线接口采用控制器 SJA1000 与收发器 MCP2551，RS232 接口电路采用 MAX232 芯片，硬件结构如图 8-14 所示。为了增加转换卡的可靠性和抗干扰性，可以在 SJA1000 与 MCP2551 之间以及微控制器与 MAX232 之间增加光耦隔离，还可以增加一些 LED 灯来指示转换卡的工作状态，便于管理和维护。

转换卡硬件主要有两大部分：RS232 接口电路和 CAN 接口电路。RS232 接口电路由 AT89C52、MAX232 及电容构成，负责该卡与计算机的通信功能，电路如图 8-15 所示。

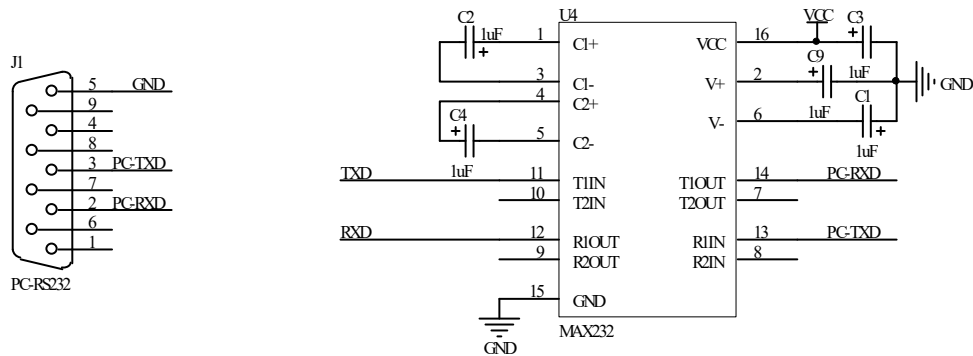


图 8-15 RS232-CAN 转换卡 RS232 接口电路部分

CAN 接口电路采用 AT89C52、SJA1000、MCP2551 及附加元器件构成，负责将接收到的计算机控制指令及数据传送至节点控制模块，其电路如图 8-16 所示。

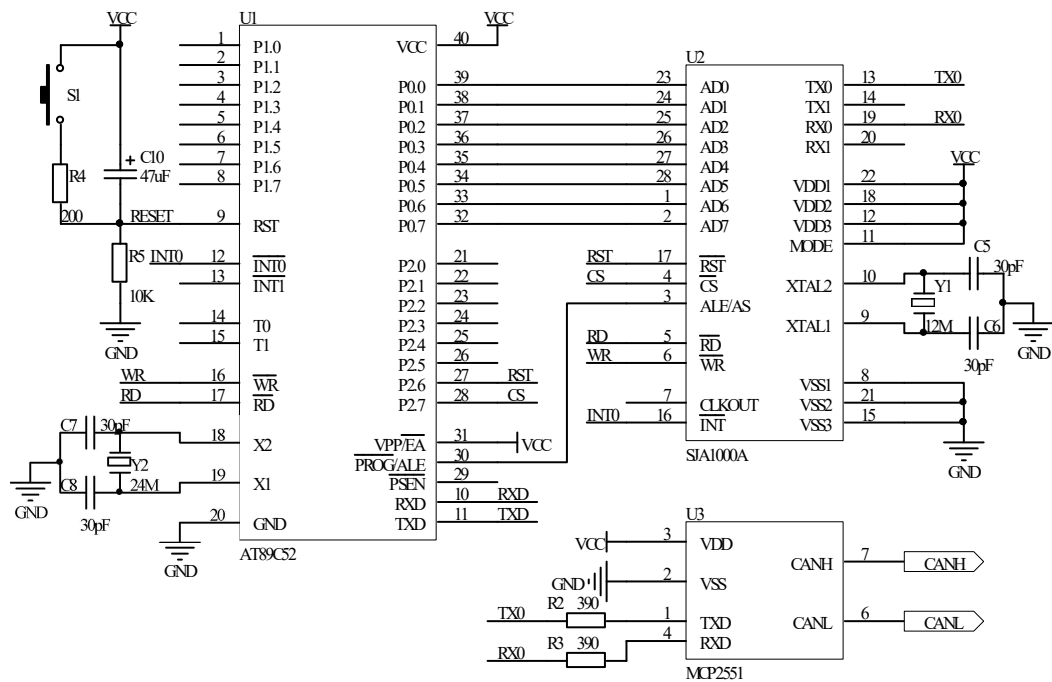


图 8-16 RS232-CAN 模块 CAN 接口电路部分

2、CAN-RS232 转换卡通信协议设计

CAN 总线标准具有物理层和数据链路层协议，以帧为单位进行数据通信，且每帧都包含帧标识符；而 RS232 本质上是一个物理层标准，以字节为单位进行数据通信，其帧格式完全由用户定义。在 CAN 总线和 RS232 上所有的信息都是以帧为单位进行传输，所以在设计帧格式时，应尽可能参考 CAN 总线数据帧模式。如果两者的帧格式类似，在程序设计时将会非常方便。RS232 帧格式结构如表 8-7 所示。

RS232 帧格式分为数据帧格式和命令帧格式，其中数据帧格式包含 RS232 网络与 CAN 网络交互数据信息，使两个网络的数据帧进行交互；命令帧格式包含 RS232 网络对转换卡的配置和查询信息，用于管理和监控。

(1) 帧头和帧尾

帧头表示每帧的开始，帧尾表示每帧的结束。为防止帧头和帧尾与帧中的其他数据冲突，采用两种方式，帧头和帧尾分别采用两个字节；在帧格式中增加了信息长度区域，即表示信息长度区域后到帧尾之间有信息长度+1 个字节非帧尾。这两种方式在一定长度上降低了冲突的概率。

(2) 数据帧标识和命令标识

数据帧标识指传输中的帧为数据帧，其中包含交互数据信息，占一个字节，采用 0x00 表示，命令帧标识指传输中的帧为命令帧，其中包含配置和查询信息，占一个字节，采用非零数值表示，不同的数值代表不同的命令。

表 8-7 RS232 帧格式定义

数据帧	帧头	数据帧标识	长度	CAN 帧信息	CAN 帧标识	CAN 帧数据	校验	帧尾
字节数	2	1	1	1	4	1~8	1	2
命令帧	帧头	命令帧标识	长度	命令参数			校验	帧尾
字节数	2	1	1	0~20			1	2

（3）信息长度

信息长度指信息长度与校验区域之间包含的字节数。信息长度在不同的帧格式中代表不同的意思，在数据帧中，信息长度指 CAN 帧信息、CAN 帧标识符、CAN 帧数据所占的字节数。在命令帧中，信息长度指命令参数所占的字节数。

（4）CAN 帧信息、CAN 帧标识符、CAN 帧数据

这 3 个区域为一个完整的 CAN 数据帧。当转换卡收到 RS232 的数据帧时，只要将这 3 个区域发送到 CAN 网络就可以了。

（5）命令参数

命令参数指在命令帧中对应命令的参数，表示一些配置信息和查询信息。

（6）校验

数据帧和命令帧中的校验都采用 CRC 方法。参与计算的字节为除帧头和帧尾的所有字节。

3、程序流程设计

如何保证数据的可靠传输是转换卡的设计中主要考虑的问题。转换卡电路设计中没有扩展外部 RAM，因此在软件设计中对 AT89S52 的内部 RAM 采取 FIFO 的管理模式以及较高的通信速率。转换卡能够接收 CAN 网络的数据帧，RS232 网络的数据帧和命令帧，因此在单片机中分配两个缓存区，分别存放接收到的 CAN 网络数据、RS232 网络数据命令。这两个缓存器采用 FIFO 方式进行管理。如果 CAN 网络数据 FIFO 存在完整的数据包，则向 RS232 网络数据 FIFO 转发；如果 RS232 网络数据命令 FIFO 存在完整的数据包，则向 CAN 网络数据 FIFO 转发；如果 RS232 网络数据命令 FIFO 存在完整的命令包，则响应该命令，并反馈结果。

转换卡的程序流程设计包括 CAN 总线接收终端服务程序、RS232 接收中断服务程序和主程序，其中主程序包括初始化和对各个 FIFO 的监控。

（1）主程序流程

主程序实现的功能包括系统初始化和缓冲区 FIFO 监控等，主程序流程如图 8-17 所示。

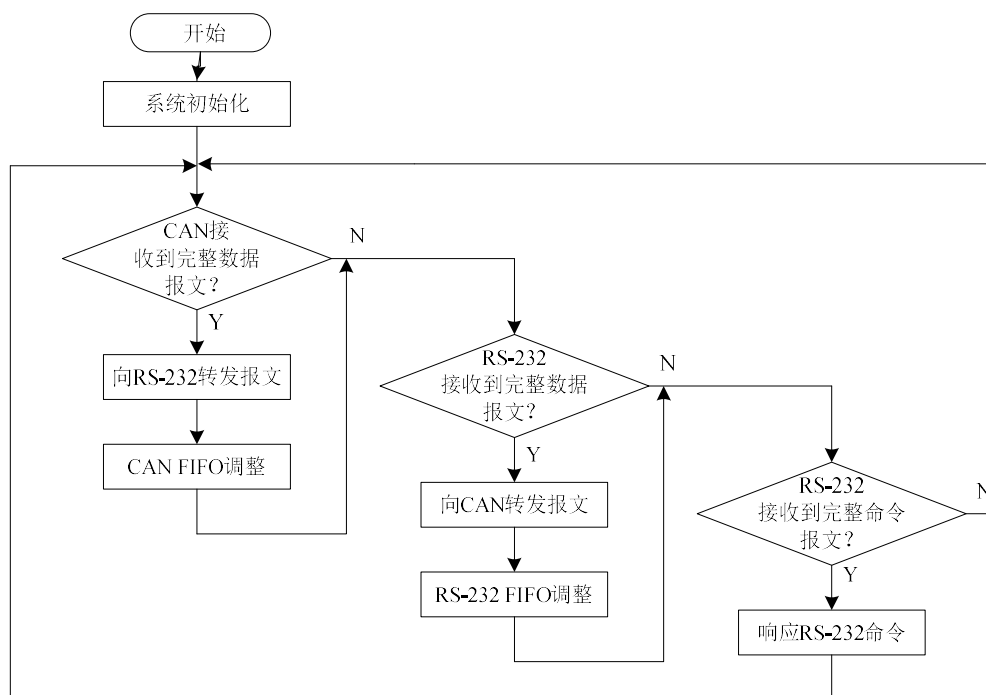


图 8-17 主程序流程图

系统初始化主要包括单片机初始化、CAN 控制器 SJA1000 的初始化以及变量初始化。单片机初始化部分主要对 RS232 通信波特率、串口接收中断等寄存器进行配置；SJA1000 初始化部分主要对 CAN 总线通信波特率、接收中断寄存器、模式寄存器、验收屏蔽寄存器、验收代码寄存器等配置；变量初始化将一些主要变量设置为默认初始值。

缓冲区 FIFO 监控主要监控接收 CAN 网络的数据 FIFO、接收 RS232 网络的数据 FIFO 和接收 RS232 网络的命令 FIFO。当接收 CAN 网络的数据 FIFO 有完整的数据报文后，就启动 RS232 发送程序，将数据报文转发至 RS232 网络；当接收 RS232 网络的数据 FIFO 有完整的数据报文后，就启动 CAN 发送程序，将数据报文转发至 CAN 网络；当接收 RS232 网络的命令 FIFO 有完整的命令报文后，根据命令执行相应动作，并反馈执行结果。

（2）CAN 总线接收中断服务流程

CAN 总线接收中断的服务程序功能为：判断接收 CAN 缓冲区是否有足够的剩余空间，如果接收 CAN 数据报文 FIFO 有足够空间，则将报文缓存到该缓存中；如果没有足够的空间，则报数据溢出错误，中断流程图如图 8-18 所示。

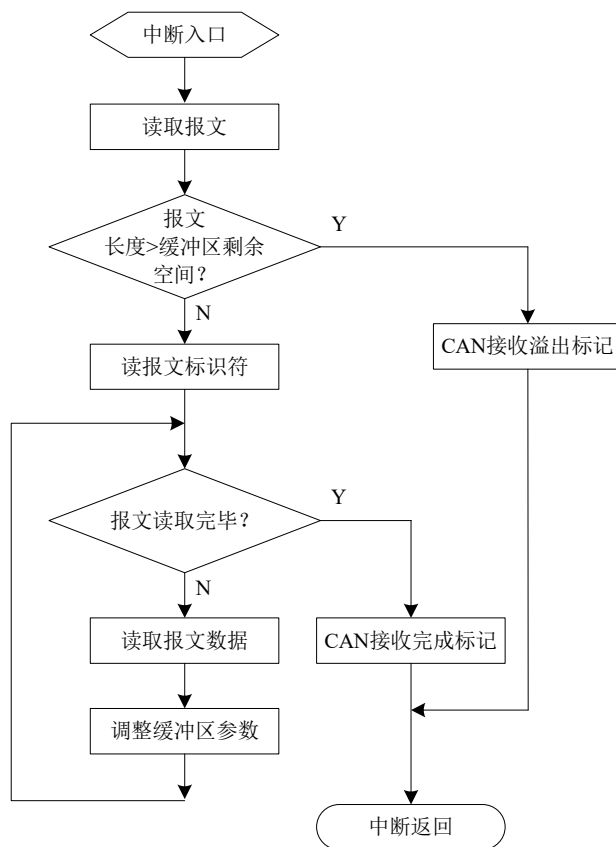


图 8-18 CAN 接收中断流程图

(3) RS232 接收中断服务程序

RS232 接收中断程序的功能为：判断接收 RS232 数据命令 FIFO 剩余空间是否 ≥ 1 ，如果有足够的剩余空间，则接收到新的报文数据，置收到新数据标志；如果剩余空间 < 1 ，没有足够的剩余空间，则给出 RS232 接收缓冲区溢出标志。具体中断流程如图 8-19 所示。

在转换卡设计中，不要将功能设计的太复杂，其主要任务是完成物理层和数据链路层的协议转换，并将过滤的报文以最快的速度转发。对于通信流量和防冲突方面的控制，应当由通信双方应用层来完成。

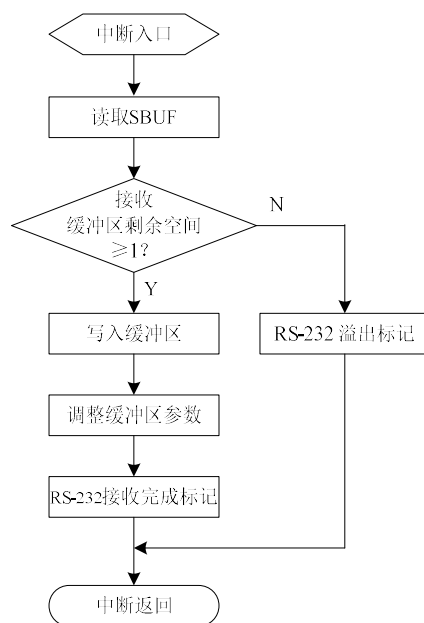


图 8-19 RS232 接收中断流程图

8.5 基于 iCAN 网络的分布式测控系统设计

本节以广州致远电子有限公司的 iCAN 教学实验开发平台为例，介绍基于 iCAN 的网络分布式测控系统设计。

8.5.1 iCAN 协议简介

iCAN 协议（Industry CAN-bus Application Protocol）由广州致远电子有限公司开发并具有自主知识产权。iCAN 协议为基于现场总线 CAN-bus 新型应用层协议之一，具有理解简单、易于实现、实时可靠的特点。

iCAN 协议规范在充分汲取了 DeviceNet 协议和 CANopen 协议之精萃的基础上，优先保障通信数据的可靠性与实时性，以相对简单的方式进行数据通信，从而有效降低了硬件实现成本，这就是 iCAN 协议的巨大优势。

iCAN 协议的规范化主要体现在以下关键因素上：

- CAN 报文的分配；
- 数据通信的实现；
- 网络管理机制；
- 设备建模。

通过上述核心技术问题的有效解决，一方面可以保证 iCAN 系统的高通信效率和高数据可靠性，使基于 iCAN 协议的各个 CAN 总线功能设备能够连接成一个有机的整体网络；另一方面，通过对 iCAN 协议在设备建模方面的规范化，可以实现产品的描述标准化与电子化，同时使 iCAN 协议具有可延续性发展空间，保障联网产品在通信协议方面的一致性。iCAN 协议主要用于中、小型现场总线网络测控系统。

8.5.2 系统组成

基于 iCAN 网络的分布式测控系统的网络拓扑采用总线式结构，分为两层：上位机监视数据统计，下层现场信号监控，如图 8-20 所示。

上层监控部分主要由两部分构成：CAN-bus 接口卡和上位机。其中 CAN-bus 接口卡一端和总线相连，完成和 CAN 总线的通信；另一端和上位机连接，完成和上位机的通信。它的主要功能是将上位机的操作信号和控制参数传送给指定的 CAN 网络节点，同时，将节点的数据传输给上位机做进一步处理系统采用的 CAN 总线智能网络通信适配器是 CAN-bus 接口卡，集成了 CAN2.0 通信协议，内置的 SJA1000CAN 控制器完成 PC 机与 CAN 现场网络之间的数据通信的控制。上位机主要是为操作人员提供友好的监控界面，包括控制参数的设置。上位机控制中心可以对测控平台进行远程实时显示和检测。利用上位机软件与控制模块进行实时通讯，将控制模块中的相关数据传输和显示在计算机终端显示器上。方便用户对每个检测信号进行监测。

现场控制层为系统的底层，主要由控制模块和现场传感器组成，完成全部的控制工作。传感器采集到的现场信息传到现场控制模块，现场控制模块根据输入信息和设定的参数进行



图 8-20 基于 iCAN 网络的分布式测控系统

运算，随即输出控制量控制相关机构进行动作或者反馈到上层监控系统。

8.5.3 系统硬件设计

iCAN教学实验开发平台由CANalyst分析仪、iCAN功能模块、传感器、运动机构以及现场总线实验板组成。主控系统由CAN-bus接口卡和PC构成：CAN-bus接口卡负责从CAN网络中收发数据，PC机与模块之间使用iCAN协议进行通信，PC端的数据处理可以由组态软件实现，也可调用iCAN库编程实现。iCAN功能模块主要包括iCAN-2404，iCAN-4017，iCAN-4050，iCAN-4400，iCAN-5303，iCAN-6202。外部信号检测设备主要包括，热电偶，热电阻传感器，接近开关，温湿度传感器、超声传感器、烟雾传感器等。硬件框图如图8-21所示。

1、PC-CAN 接口卡

CANET-E 是周立功公司开发的一款嵌入式网络适配器，如图8-22所示，它内部集成了CAN-Bus 接口和EtherNet 接口以及TCP/IP 协议栈用户利于它可以轻松完成CAN网络和EtherNet 网络的互连互通进一步拓展CAN-网络的范围。

CANET-E 适配器的上下两端分别是以太网接口和 CAN 接口。首先说明一下以太网接口，以太网接口一端有两盏 LED 和两个 RJ45 接口如图上方 Status 对应的 LED 显示以太网的网络状态黄绿闪烁表示发送数据到以太网黄红闪烁表示接收到以太网的数据 Link 对应的 LED 显示以太网网线以经正确连接绿色表示网线已经正确连接黄绿闪烁表示以太网上接收的数据溢出两个 RJ45 接口表示用户可以使用不同的网线平行线或交叉线与 CANET-E 适配器相连只要正确连接 Link LED 就会显示绿色。

下方可以看出CAN 接口有10 个引出接线柱实际有用的只要4 个接线柱，CAN_L、CAN_H 的功能就很清晰了，就是连接CAN-Bus，+Vs 连接外部直流电源的正极，0V 连接外部直流电源的地外接电源，电压范围是30V~9V。

Power 灯在空闲状态下每5 秒绿色亮一下，CAN 有数据接收时绿灯闪烁，CAN 有数据发送时红灯闪烁。

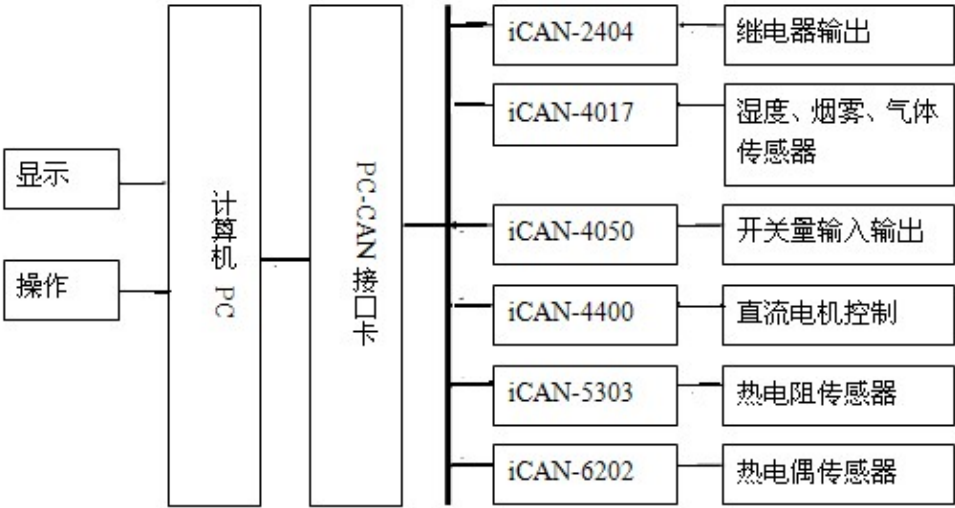


图 8-21 硬件框图



图 8-22 CANET-E 外观示意图

2、iCAN 系统功能模块

(1) iCAN-4050 模块

iCAN-4050 DI/DO 功能模块为数字量输入/输出模块，用于采集数字量输入信号，并可以输出数字量信号，控制外部电子设备，外观如图 8-23 所示。iCAN-4050 功能模块的数字量输入和输出通道采用非隔离设计。iCAN-4050 模块具有 8 路数字量输入通道，8 路数字量输出通道。

1) 主要技术指标



图 8-23 iCAN-4050 外观示意图

- 单电源供电，供电电压：+10~+30V DC；
- 输入通道数：8 路数字量信号；
- 输出通道数：8 路数字量信号；
- 高电平信号（数字 1）：+3.5 ~+30V；
- 低电平信号（数字 0）：≤+1V；
- 数字量输出信号：集电极开漏输出，最大负载电压+30V，最大负载电流 30mA。

2) 模块接口说明

iCAN-4050 模块接口如图 8-24 所示，模块内部各接线端子、拨码开关、跳线器以及指示灯功能说明如下：

- SW1: 8 位拨码设置开关，用于设置节点地址以及 CAN 总线通信波特率；
- RL1: 电源指示灯，用于指示模块的电源工作状态；
- SL2: 网络通信指示灯，用于指示模块的通信状态；
- JP1: DC 电源、CAN 通信接口以及数字量输入信号通道接线端子；
- JP2: 数字量输入信号通道和数字量输出信号通道接线端子。

下面介绍模块接线端子的引脚分配，iCAN-4050 模块的接线端子 JP1、JP2 引脚定义说明如下：

- 模块电源接线端子

电源必须为+10~+30V 的直流电源，电源引脚包括+VS 和 GND 两个接线引脚。

+VS: 连接电源正端（+10~+30V DC）；

GND: 连接电源负端。

- CAN 总线通信接线端子

CAN 通信接线端子包括 CAN_L、CAN_H、Res- 以及 Res+ 4 个接线端子

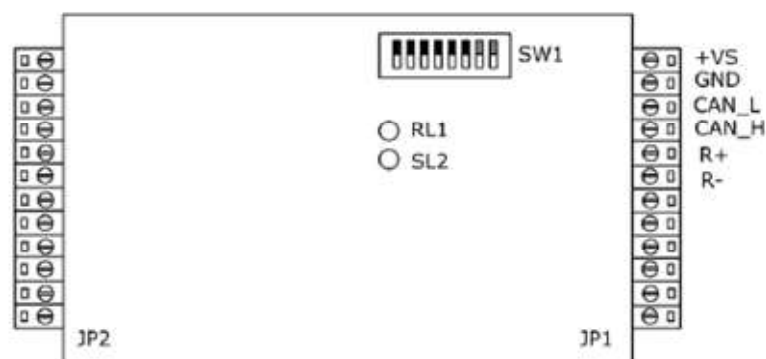


图 8-24 iCAN-4050 模块接口示意图

CAN_L: 连接 CAN 通信线的 CAN_L 信号线；

CAN_H: 连接 CAN 通信线的 CAN_H 信号线；

Res-、Res+: 连接 CAN 网络终端匹配电阻；

仅当模块处于网络终端位置时，Res-、Res+之间需要连接终端匹配电阻。

- 模块输入/输出端口接线端子

DIN0~7: 连接数字量输入通道 0~7 信号正端；

DOUT0~7: 连接数字量输出通道 0~7 信号正端；

COM: 连接数字量输入/输出信号的参考地。

● 未用端子 NC

3) iCAN-4050 数字量输入

数字量输入的含义是指这种类型的输入信号只有简单的两种状态：高电平或低电平，也可以理解为开（ON）或者关（OFF）两种状态。

现场的数字量输入信号主要为开关触点信号和电平信号。在一般的工业控制场合，+24V 直流电平信号采用较多。

● 数字量输入状态

iCAN-4050 模块具有 8 路的数字量输入通道。iCAN-4050 模块可以采集电压类型的数字量输入信号或者触点型输入信号。

在 iCAN-4050 模块中，输入信号逻辑定义如表 8-7 所列。

表 8-7 输入信号定义

输入信号类型		输入信号定义
电压型数字量信号	高电平信号	状态 1，电压范围是+3.5 ~+30V
	低电平信号	状态 0，电压范围是≤+1V
无源触点型数字量信号	开路触点信号（OFF 状态）	状态 1
	闭合触点信号（ON 状态）	状态 0

当输入信号电平≥3.5V 时，模块即认为输入为高电平信号（状态 1）；当输入电平信号≤+1V 时，模块即认为输入为低电平信号（状态 0）。当输入为无源触点型输入信号，对于闭合信号，模块认为输入信号为状态 0；对于开路触点信号，模块认为输入信号为状态 1。

注意：模块的数字量输入信号电压值最高不能超过+30V，否则可能会使模块受到损坏。

● 数字量输入的接线

iCAN-4050 模块的输入信号主要包括电压型和无源触点型开关量信号，它们的接线方式有所不同，如图 8-25 所示。

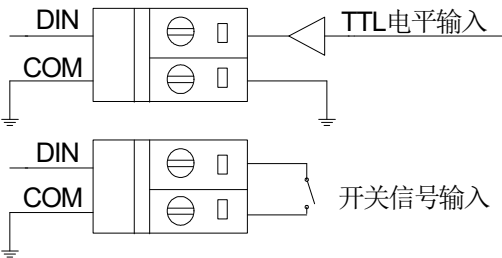


图 8-25 数字量信号输入接线示意图

在连接电压型输入信号时，注意信号输入的正端与 DIN 端子脚相连接，输入信号的负端与 COM 端子脚相连接。如果是多路输入信号，则输入信号的正端分别与不同的 DIN 端子脚相连接，所有输入信号的负端与 COM 端子脚相连接。电压型数字量信号接线时要注意信号极性，以免接反。

在连接无源触点信号时，则只需注意触点开关的一端与 DIN 端子脚相连接，触点开关的另外一端与 COM 端子脚相连接。

- 数字量输入的测试电路

数字量输入测试采用的电路如图 8-26 所示。

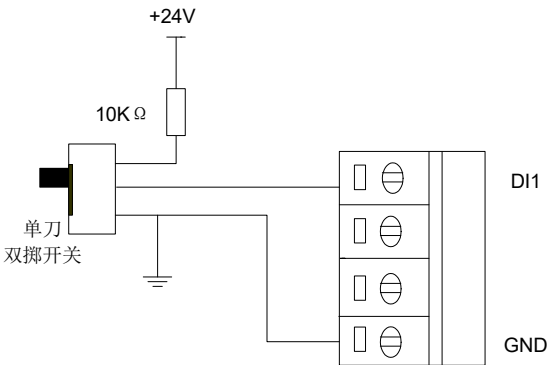


图 8-26 数字量输入测试电路

在图 8-26 中，单刀双掷开关的公共端连接到 iCAN-4050 输入通道 DI 1 上，单刀双掷开关的另外两端分别连接电源和地。当拨动开关时，即可使输入端连接到高电平或者低电平。

4) iCAN-4050 数字量输出

数字量输出的含义是指这种类型的输出信号只有简单的两种状态：高电平或低电平，也可以理解为开(ON)或者关(OFF)两种状态。

iCAN-4050 模块具有 8 路的数字量输出通道。iCAN-4050 模块输出为晶体管开漏输出，可以向外提供电压型数字量输出信号。iCAN-4050 模块输出的最大负载电压为 +30 V，最大负载电流为 30mA。在应用 iCAN-4050 的数字量功能时，需要在输出端口连接负载以及上拉电源。

- 数字量输出的接线方式

数字量输出接线示意图如图 8-27 所示。

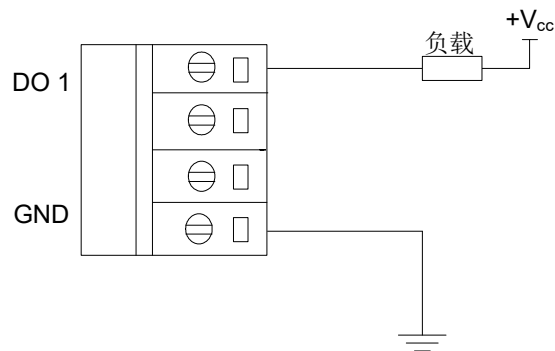


图 8-27 数字量输出接线示意图

iCAN-4050 输出通道在使用时必须连接上拉电阻。iCAN-4050 的 DOUT 端子脚与用户提供的上拉电阻连接，GND 端子脚与用户提供的信号地相连接。

iCAN-4050 模块的输出信号驱动继电器接线方式如图 8-28 所示。

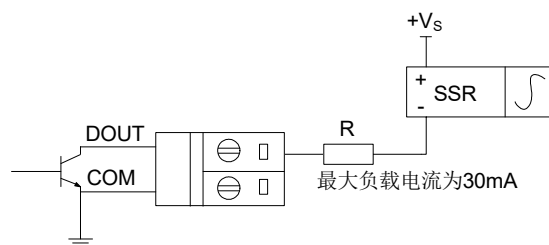


图 8-28 数字量输出驱动继电器接线示意图

● 数字量输出的测试电路

数字量输出的测试电路如图 8-29 所示。

图 8-30 中电源电压为 +24 V，通过控制数字量通道 DO1 的输出状态可达到使 LED1 亮灭的效果。

假设流过发光二极管 LED1 的电流为 I ， I 只要控制在 5~10 mA，就足够点亮 LED。电流的具体算法也很简单，只要用所提供的电源电压减去 LED 和内部晶体管所产生饱和管压降，所得到的电压除以用户外接的电阻就可以得到流过 LED 的电流了。在数字量输出测试电路图 8-29 中电压为 24 V，外接电阻为 3K Ω ，大概算得流过 LED 的电流为 8 mA。

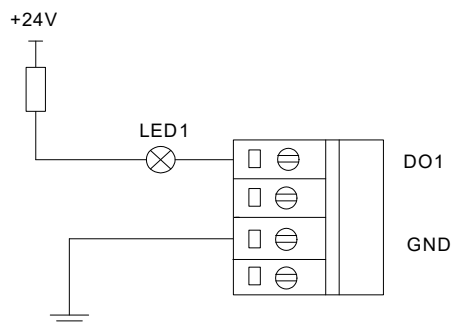


图 8-29 数字量输出测试电路图

(2) iCAN-4400模块

iCAN-4400模拟量输出模块提供电流或者电压输出信号。iCAN-4400模块具有4路模拟量输出通道，可输出1~5V电压或者4~20mA电流信号。模块在工作时，iCAN主站设备通过I/O数据通讯，将输出数据传送给单片机后，通过光电隔离送到DA模块输出。输出信号类型可以通过跳线器选择电压输出或者电流输出。

1) 主要技术指标

- 单电源供电，供电电压：+10V~+30V DC；
- 输出通道数：4路；
- 输出信号：1~5V、4~20mA；
- DA转换分辨率：12位；
- 输出精度：±1.0%；
- 电流输出负载能力：小于500Ω；
- 隔离电压：1000V DC（模块供电、信号输入）。

2) 模块接口说明

iCAN-4400模块具有接口及设置开关如上图8-30所示，模块内部各接线端子、拨码开关、跳线器以及指示灯功能说明如下：

- SW1：模块CAN波特率以及MAC ID设置开关；
- RL1：电源指示灯；
- SL2：网络通讯指示灯；
- JP1：+10V~+30V DC电源、iCAN通讯接口；
- JP2：模拟量输出信号接线端子；
- JP400~JP403：0~3通道的输出电压或者输出电流选择跳线器；

iCAN-4400模块的接线端子JP1、JP2引脚定义如图 8-31所示，说明如下：

- +VS：+10V~+30V DC电源正端；

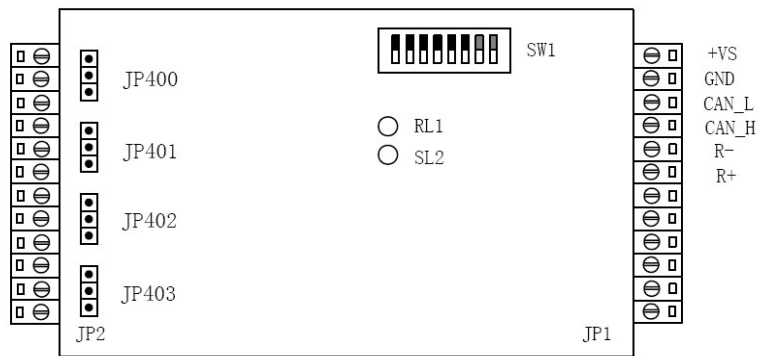


图 8-30 iCAN-4400 模块接口示意图

- GND : +10V~+30V DC电源负端;
- CAN_L : CAN通讯信号CAN_L端 ;
- CAN_H : CAN通讯信号CAN_H端;
- Res- : 接CAN网络终端匹配电阻;
- Res+ : 接CAN网络终端匹配电阻;
- +24V : 电流输出端;
- OUT0~OUT3: 模拟量信号输出端;
- AGND: 电压输出端;
- NC : 未用端子。

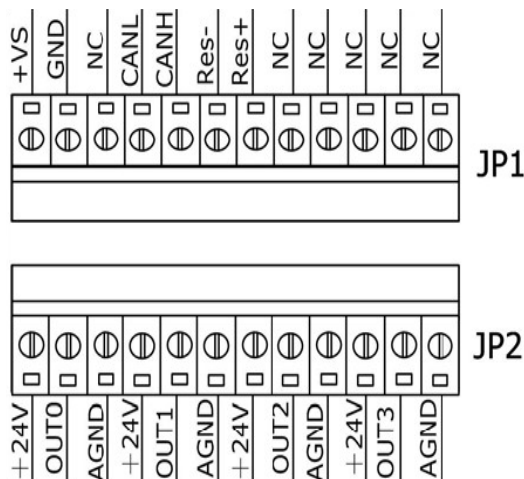


图 8-31 接线端子定义

(3) iCAN -4017模块

iCAN -4017AI功能模块用于采集模拟量输入信号。iCAN -4017模块具有8路模拟量输入通道，16位分辨率。模块在工作时，将输入的电压信号或者电流信号经多路开关、AD转换后经光耦隔离模块送入单片机，通过CAN总线通讯将输入的模拟量信号状态传送到网络中

的主控设备。

1) 主要技术指标

- 单电源供电，供电电压：+10V~+30V DC；
- 输入通道数：6路差分输入，2路单端输入；
- 输入信号范围：±10V (默认)、±5V、±2.5V、±1V、±500mV、±150mV
- 电流输入：±20mA（需外接125Ω精密电阻）
- AD转换分辨率：16位
- 转换速率：2次/每秒（8通道每次）
- 输入低通滤波、过压保护
- 隔离电压：1000V DC（模块供电、信号输入）

2) 模块接口说明

iCAN -4017模块具有接口及设置开关如上图8-32所示，模块内部各接线端子、拨码开关、跳线器以及指示灯功能说明如下：

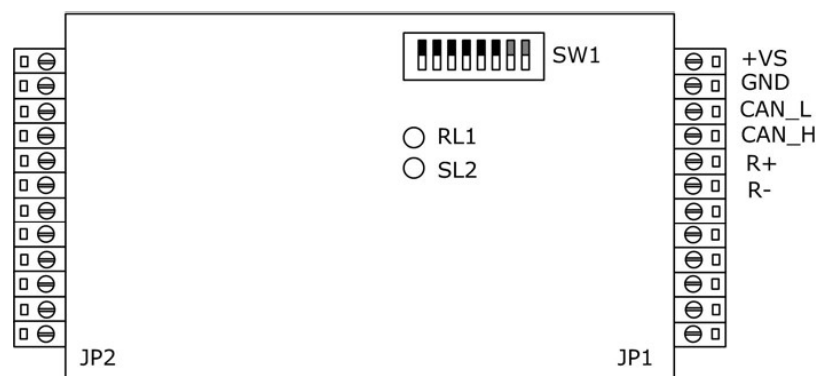


图 8-32 iCAN-4017 模块接口示意图

- SW1：模块CAN波特率以及MAC ID设置开关
- RL1：电源指示灯
- SL2：网络通讯指示灯
- JP1：DC电源、CAN通讯接口以及模拟量输入信号通道接线端子
- JP2：模拟量输入信号通道接线端子

iCAN -4017模块的接线端子JP1、JP2引脚定义如图8-33所示。各引脚说明如下：

- +VS：+10V~+30V DC电源正端
- GND：+10V~+30V DC电源负端

- CAN_L : CAN通讯信号CAN_L端

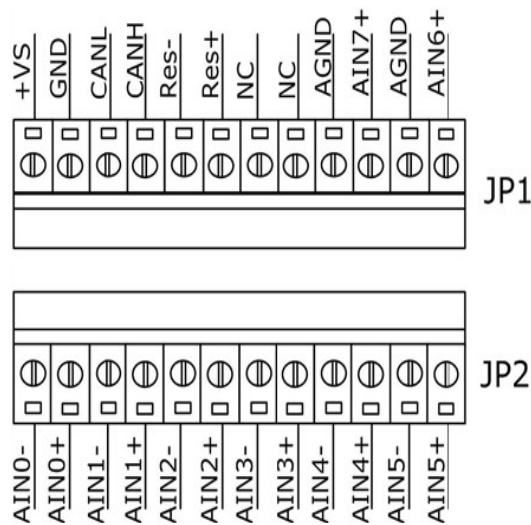


图 8-33 接线端子定义

- CAN_H : CAN通讯信号CAN_H端
- Res- : 接CAN网络终端匹配电阻
- Res+ : 接CAN网络终端匹配电阻
- Ain0+~Ain7+ : 接模拟量输入通道0~7 信号正端
- Ain0-~Ain5- : 接模拟量输入通道0~7 信号负端
- AGND: 模拟量输入通道6、7输入参考地
- NC : 未用端子

(4) iCAN -2404控制模块

iCAN-2404继电器输出模块可用于工业现场，提供继电器输出通道。iCAN-2404继电器输出模块有4路具有自保持功能的继电器输出通道。模块在工作时，iCAN主站设备通过I/O数据通讯，将继电器控制数据传送给单片机后，通过光电隔离后输出驱动继电器。

1) 主要技术指标

- 单电源供电，供电电压：+10V~+30V DC；
- 输出通道数： 4路
- 触点形式：2a 或2b（触点输出状态自保持）
- 触点控制：“1”吸合，“0”断开
- 触点容量：DC： 24VDC/1A
- AC： 220VAC/0.5A

- 触点寿命： 5×10^5
- 隔离电压：1000V DC（模块供电、信号输入）

2) 模块接口说明

iCAN-2404模块具有接口及设置开关如图8-34所示，模块内部各接线端子、拨码开关、跳线器以及指示灯功能说明如下：

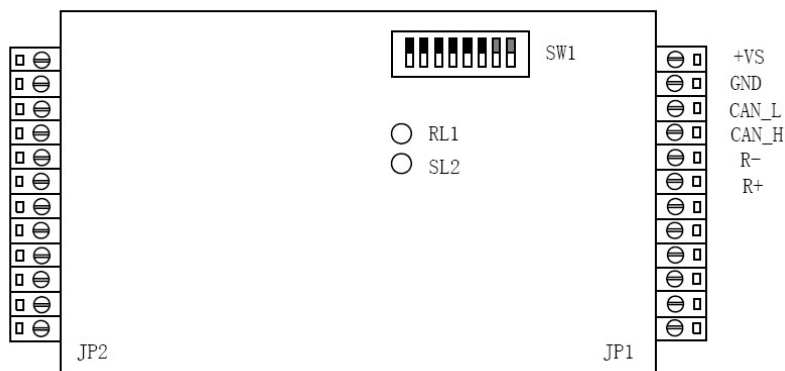


图 8-34 iCAN-2404 模块接口示意图

- SW1：模块CAN波特率以及MAC ID设置开关
- RL1：电源指示灯
- SL2：网络通讯指示灯
- JP1：+10V~+30V DC电源、iCAN通讯接口以及继电器接线端子
- JP2：继电器接线端子

iCAN-2404模块的接线端子JP1、JP2引脚定义如图 8-35所示。说明如下：

- +VS：+10V~+30V DC电源正端
- GND：+10V~+30V DC电源负端
- CAN_L：CAN通讯信号CAN_L端
- CAN_H：CAN通讯信号CAN_H端
- Res-：接CAN网络终端匹配电阻
- Res+：接CAN网络终端匹配电阻
- DO0A 1~DO0A 2：继电器0输出A接线端子
- DO0B 1~DO0B 2：继电器0输出B接线端子
- DO1A 1~DO1A 2：继电器1输出A接线端子
- DO1B 1~DO1B 2：继电器1输出B接线端子
- DO2A 1~DO2A 2：继电器2输出A接线端子

- DO2B 1~ DO2B 2 ：继电器2输出B接线端子
- DO3A 1~ DO3A 2 ：继电器3输出A接线端子
- DO3B 1~ DO3B 2 ：继电器3输出B接线端子
- NC ： 未用端子

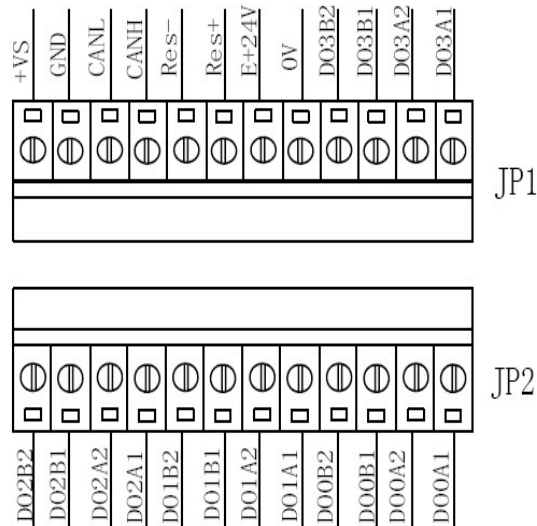


图 8-35 接线端子定义

(5) iCAN-6202模块

iCAN-6202热电偶输入模块用于采集温度。iCAN-6202模块具有2路热电偶输入通道，支持8种标准化热电偶类型，温度值分辨率0.1℃，支持定时循环传送及温度超限报警。额外的2通道数字量输出，既可用于指示模块工作状态也可由用户自行控制。模块工作时，周期性地输入电压信号经低通滤波、AD转换、光电隔离后送入单片机，单片机对AD转换值经非线性校正及冷端补偿等处理后得到温度值，通过CAN总线通讯将温度值传送到网络中的主控设备。

1) 主要技术指标

- 单电源供电，供电电压：+10V~+30V DC；
- 输入通道数： 2路；
- 输出通道数： 2路，可独立配置为输入通道状态指示模式或用户控制模式；
- 输出通道类型：集电极开漏输出，最大负载电压+30V，最大负载电流30mA；
- 支持的热电偶类型及测温范围： J型 -210℃~1200℃、K型 -200℃~1370℃、E型 -100℃~1000℃、T型 -200℃~400℃、N型 -200℃~1300℃、B型 650℃~1800℃、R型 0℃~1750℃、S型 0℃~1760℃；
- 温度值分辨率： 0.1℃；
- 热电偶冷端补偿精度： ±1℃；
- 转换速率： 4次/秒（2通道/次）；

- 定时循环传送时间间隔：最小值 10毫秒、最大值 2.55秒；
- 温度超限报警。

2) 模块接口说明

iCAN-6202模块具有接口及设置开关如上图8-36所示，模块内部各接线端子、拨码开关、跳线器以及指示灯功能说明如下：

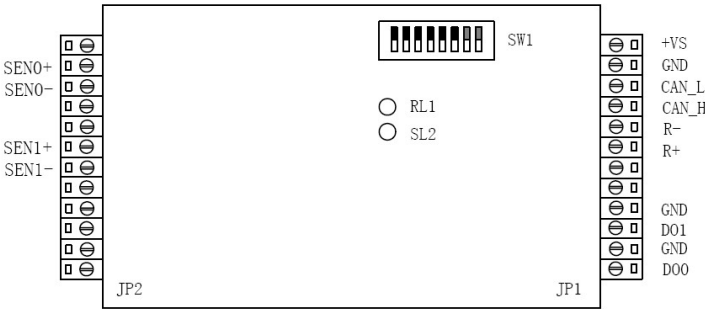


图 8-36 iCAN-6202 模块接口示意图

- SW1 ： 模块CAN波特率以及MAC ID设置开关；
- RL1 ： 电源指示灯；
- SL2 ： 网络通讯指示灯；
- JP1 ： DC电源、CAN通讯接口以及数字量输出信号通道接线端子；
- JP2 ： 热电偶输入通道接线端子。

iCAN-6202模块的接线端子JP1、JP2引脚定义如图8-37所示。说明如下：

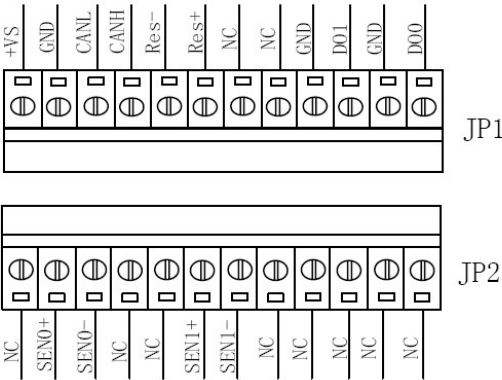


图 8-37 接线端子定义

- +VS ： +10V~+30V DC电源正端；
- GND ： +10V~+30V DC电源负端；
- CAN_L ： CAN通讯信号CAN_L端；
- CAN_H ： CAN通讯信号CAN_H端；

- Res—：接CAN网络终端匹配电阻；
- Res+：接CAN网络终端匹配电阻；
- DO0：数字量输出通道0；
- DO1：数字量输出通道1；
- SEN0+：通道0热电偶输入正；
- SEN0—：通道0热电偶输入负；
- SEN1+：通道1热电偶输入正；
- SEN1—：通道1热电偶输入负；
- NC：未用端子。

3) 热电偶的接线方法

热电偶输入信号正端连接到SEN+，输入信号负端连接到SEN—，如图 8-38所示。

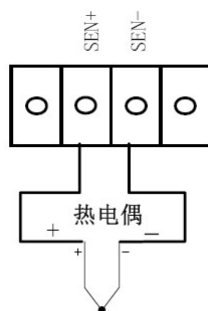


图 8-38 热电偶接线图

4) 数字量输出

iCAN-6202模块提供两路集电极开漏数字量输出DO0和DO1，最大负载电压+30V，最大负载电流30mA。由于是集电极开漏输出，用户在使用时需加上拉电阻。

数字量输出通道可工作在两种工作模式：输入通道状态指示模式、用户控制模式。

● 输入通道状态指示模式

这是模块输出通道的默认工作模式，在这种模式下，DO0用于指示通道0的工作状态，当通道0的温度超限或热电偶断线时，DO0输出为高，平时DO0输出为低；DO1用于指示通道1的工作状态，当通道1的温度超限或热电偶断线时，DO1输出为高，平时DO1输出为低。

● 用户控制模式

通过向模块写配置字可设置为用户控制模式。在这种模式下，用户可通过向DO单元（资源地址0x20）写1字节数据来控制管脚的输出状态，该字节的第0位控制DO0、第1位控制DO1。

输出通道切换为用户控制模式后、在用户向DO资源（资源节点0x20）写数据之前，输出通道输出安全值，该安全值由用户配置。

8.5.4 系统软件设计

在开始工程设计之前，先对整个工程进行剖析，以便从整体上把握工程的结构、流程、需实现的功能及如何实现这些功能，本章示例工程的软件主要从硬件测试、OPC 设置、组态工程的建立三个方面来进行设计。

1、iCAN 模块测试方法

对于 iCAN 功能模块的测试，组要有两种上位机测试方法，一种是利用 iCANTest 测试软件，另一种是利用 ZLGCANTest 测试软件。掌握了这两种方法后，用户可以根据提供的 iCAN 协议库及底层驱动开发出基于 API 及 ZOPC_Server 接口的应用平台，以适用于不同的应用场合。

以 iCAN4050 模块为例，测试所需要设备为 PC 机、CAN-bus 接口卡和 iCAN-4050 模块，接线如图 8-39 所示。

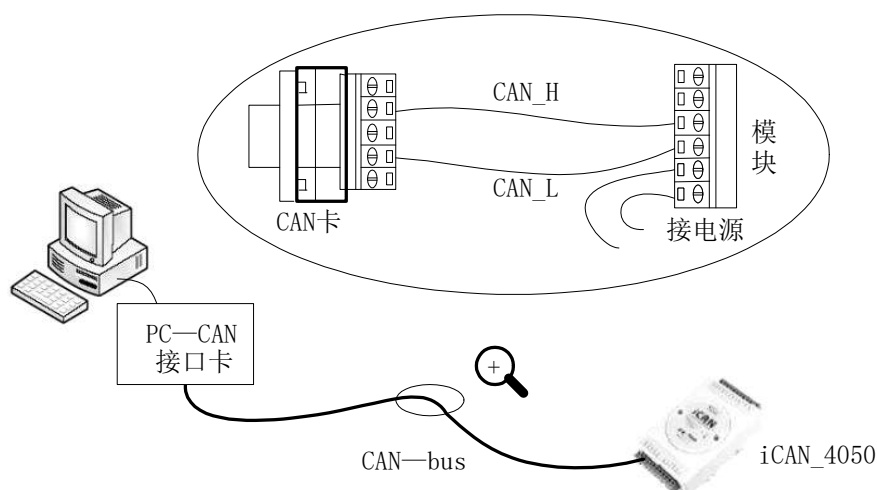


图 8-39 iCAN-4050 测试接线示意图

在测试之前，将 iCAN-4050 功能模块的地址开关的第一位拨到 OFF 位置，其余的开关都拨到 ON 位置，此时模块的 MACID 为 1，波特率设定值为 0x00, 0x1C，将模块的电源线和 CAN 通信线连接好，并将上位机 CAN 通信线与模块的 CAN 通信线相连。上电后会看到 iCAN-4050 模块 MNS 指示灯经历了红灯亮→红灯灭→绿灯亮的过程。

(1) iCANTest 测试示例

● 系统配置

单击“系统配置”按钮，设置主站波特率为 500 Kbps，主站定时循环参数为 100ms，单击确定按钮，如图 8-40 所示。

● 搜索模块

在设置好系统配置后，单击“搜索”按钮，可以得到从站信息：序号为 0；设备型号为

iCAN-4050; MAC 地址为 1, 如图 8-41 所示。

- 启动 CAN 卡

当单击启动 CAN 卡时, PC 机将以系统配置参数来初始化上位机 CAN 节点。单击“启动”按钮后, “上线”按钮将被激活, 如图 8-42 所示。



图8-40 系统配置窗口



图 8-41 搜索示意图



图 8-42 启动主站

- 模块上线

单击“上线”按钮, 将会看到模块的指示灯在闪烁, 如图 8-43 所示, 可以观察模块的输入端口状态; 同时单击代表输出端口的按钮, 可以控制模块的输出状态。

- 设置安全输出

iCAN-4050 模块在上线的情况下。可以由用户直接控制，当模块突然掉线或模块刚上电运行时，模块的输出采用安全值进行输出。单击图 8-43 中“配置属性”按钮，出现如图 8-44 所示界面，用户可以根据需要设置 iCAN-4050 模块的安全输出值。



图8-43 iCAN-4050模块上线



图 8-44 iCAN-4050 安全值设置

(2) ZLGCANTest 测试示例

- 系统配置

首先选择CAN-bus接口卡类型，单击“打开设备”按钮，设置定时器0为00，定时器1为1C，此时的波特率被设置为500 kbps，单击“确定”按钮。如图8-45所示。



图8-45 ZLGCANTest配置示意图

● 系统启动

单击主界面上“启动 CAN”按钮，并按图 8-46 选择发送格式为“正常发送”，设置帧类型为扩展帧。



图 8-46 启动 CAN

● 建立连接

主界面中帧 ID 为 24f7，数据部分为 3 个字节：00 00 00，第 1 个字节表示分段码，第 2 个字节为主站 ID，第 3 个字节为设置的定时参数。当第 3 个字节为 00 时，从站的状态一直处于连接状态，单击“发送”按钮。在正常的情况下，会返回一帧数据，如图 8-47 所示。



图 8-47 建立连接

● 读输入端口测试

根据读命令操作时的 iCAN 协议格式，主界面中帧 ID 为 2200，数据填充为 00 01。其中数据的第 1 个字节为分段码，第 2 个字节为读数据长度，如图 8-48 所示。



图 8-48 读输入端口

● 写输出端口测试

根据写命令操作时的 iCAN 协议格式，主界面中帧 ID 为 2120，数据填充为 00 01。其中数据的第 1 个字节为分段码，第 2 个字节为输出控制值，如图 8-49 所示。



图 8-49 写输出端口

● 设置安全输出

根据 iCAN 协议中设置安全值的格式，主界面中帧 ID 为 21f9，数据填充为 00 20 01。其中数据的第 2 个字节为资源子节点地址，第 3 个字节为安全输出值。具体设置如图 8-50 所示。



图 8-50 设置安全输出值

以上是利用 ZLGCANTest 简单测试 iCAN-4050 模块的功能，当然用户可以在深入了解 iCAN 协议的基础上，测试其他功能。

● 删除连接

当操作删除连接后，模块的输出将以安全值输出。删除连接操作见图 8-51 所示。



图 8-51 删除连接

2、ZOPC 设置

安装并运行ZOPC_Server 2.50 软件。单击菜单“设备操作→iCAN→添加新设备→USBCAN2”，弹出“属性-USBCAN2”对话框，按表8-8的参数在如图 8-52所示的界面上进行主站配置。



图8-52 ZOPC 设备属性对话框

表8-8 ZOPC 设备属性的配置

配置项	数值	说明
设备索引号	0	在 1 台 PC 上可能会装有多多个相同型号的主站设备,该索引号用于索引这些设备
设置运行 iCAN	iCAN0	某些主站设备可能带有多多个 CAN 通道,在此项选择要操作的通道
波特率	500K	设置总线波特率
主站 ID	0	设置主站 ID
数据刷新时间	100ms	设置总线刷新周期

单击“添加设备”按钮，这时在“iCAN”面板上将会显示主站设备，如图8-53所示。



图 8-53 ZOPC 添加设备对话框

在“iCAN”面板上单击选中“iCAN0”节点，然后单击菜单“设备操作→iCAN→添加新从站”，在弹出的“Slave 属性”对话框中按照表8-8参数添加从站设备，如图8-54所示。



图8-54 ZOPC 从站设备属性对话框

点击“关闭”按钮，在ZOPC_Server 的iCAN 面板上将会出现如所示的从设备及其输入输出数据项，如图8-55所示。

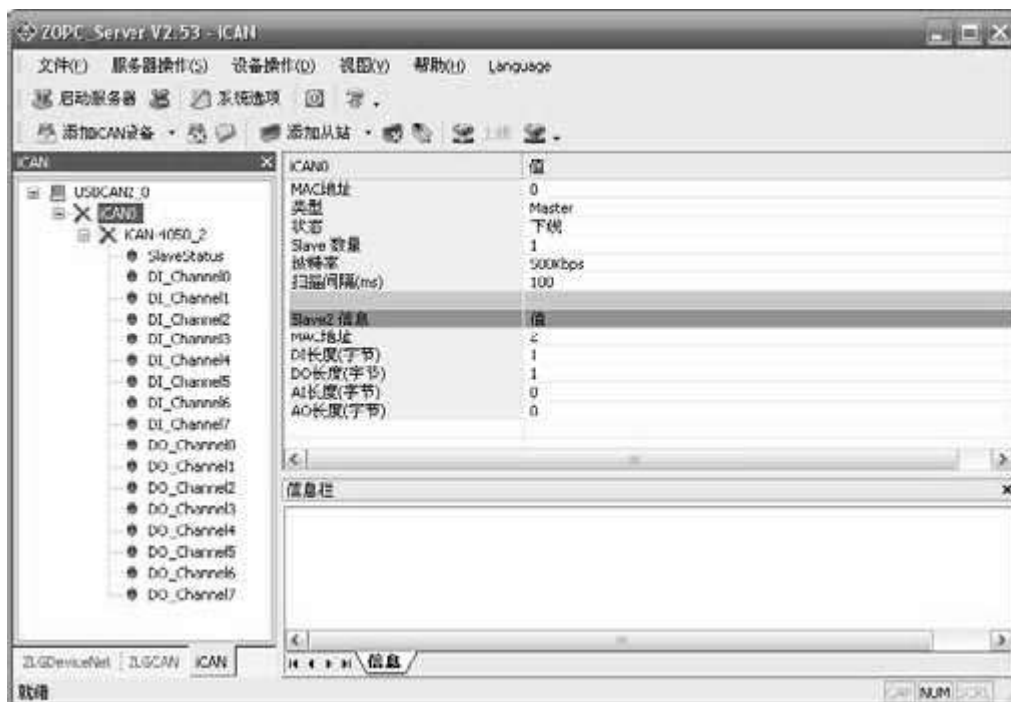


图8-55 ZOPC 从设备及其数据状态窗口

点击“服务器操作→启动服务器”，然后在iCAN 面板上点选USBCAN2_0 节点的子节点iCAN0，单击右键，在弹出菜单选择“上线”。如果设备连接无误，“iCAN0”节点的子节点，即该网络中的从站设备的图标会由变为形。此时OPC 服务器的设置已经完成，OPC 的客户端可以从服务器中读写数据。

3、组态工程设计

本设计以 MCGS 组态软件实现上位机监控软件的设计，MCGS 可以直接与前面设置的 ZOPC 进行连接。

(1) 建立 MCGS 工程

鼠标单击文件菜单中“新建工程”选项，如果MCGS 安装在D: 盘根目录下，则会在D:\MCGS\WORK\下自动生成新建工程，默认的工程名为：“新建工程X.MCG”(X表示新建工程的序号，如：0、1、2 等)。选择文件菜单中的“工程另存为”菜单项，弹出文件保存窗口。在文件名一栏内输入“基于iCAN网络的测控系统”，点击“保存”按钮，工程创建完毕，如图8-56所示。



图8-56 MCGS 启动窗口

(2) 制作工程画面

制作工程画面的步骤如下：

在“用户窗口”中单击“新建窗口”按钮，建立“窗口0”。选中“窗口0”，单击“窗口属性”，进入“用户窗口属性设置”。将窗口名称改为：主控窗口；窗口位置选中“在屏幕中间显示”，其他不变，单击“确认”。在“用户窗口”中，选中“主控窗口”，点击右键，选择下拉菜单中的“设置为启动窗口”选项，将该窗口设置为运行时自动加载的窗口。选中“主控窗口”窗口图标，单击“动画组态”，进入动画组态窗口，开始编辑画面，如图8-57所示。



图8-57 MCGS 中设置启动窗口对话框

通过工具箱可以在画面中添加控件、元件、对象等，如图8-58所示。

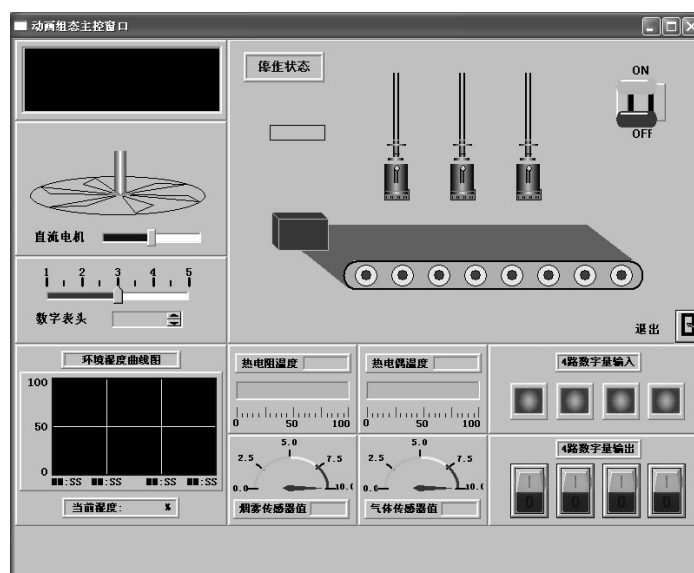


图 8-58 组态界面

(3) 定义数据对象

下面以数据对象“开关量输入”为例，介绍一下定义数据对象的步骤：

单击工作台中的“实时数据库”窗口标签，进入实时数据库窗口页，如图 8-59 所示。单击“新增对象”按钮，在窗口的数据对象列表中，增加新的数据对象，系统缺省定义的名称为

“Data1”、“Data2”、“Data3”等。选中对象，按“对象属性”按钮，或双击选中对象，则打开“数据对象属性设置”窗口。将对象名称改为：开关量输入 0；对象类型选择：开关型；单击“确认”。按照此步骤，设置其他数据对象。

主控窗口 设备窗口 用户窗口 实时数据库 运行策略				
名字	类型	注释		
data	数值型			
InputETime	字符型	系统内建数据对象		
InputSTime	字符型	系统内建数据对象		
InputUser1	字符型	系统内建数据对象		
InputUser2	字符型	系统内建数据对象		
光电传感器	开关型			
光电开关	开关型			
接近开关0	开关型			
接近开关1	开关型			
接近开关2	开关型			
开关量输出0	开关型			
开关量输出1	开关型			
开关量输出2	开关型			
开关量输出3	开关型			
开关量输入0	开关型			
开关量输入1	开关型			
开关量输入2	开关型			
开关量输入3	开关型			

图8-59 数据对象

(4) 动画连接

所谓动画连接，实际上是将用户窗口内创建的图形对象与实时数据库中定义的数据对象，建立起对应的关系，在不同的数值区间内设置不同的图形状态属性（如颜色、大小、位置移动、可见度、闪烁效果等），将物理对象的特征参数以动画图形方式来进行描述，这样在系统运行过程中，用数据对象的值来驱动图形对象的状态改变，进而产生形象逼真的动画效果。建立动画连接的操作步骤是：

- 鼠标双击图元、图符对象，弹出“动画组态属性设置”对话框；
- 对话框上端用于设置图形对象的静态属性，下面四个方框所列内容用于设置图元、图符对象的动画属性。上图中定义了填充颜色、水平移动、垂直移动三种动画连接，实际运行时，对应的图形对象会呈现出在移动的过程中填充颜色同时发生变化的动画效果；
- 每种动画连接都对应于一个属性窗口页，当选择了某种动画属性时，在对话框上端就增添相应的窗口标签，用鼠标单击窗口标签，即可弹出相应的属性设置窗口；
- 在表达式名称栏内输入所要连接的数据对象名称。也可以用鼠标单击右端带“？”号图标的按钮，弹出数据对象列表框，鼠标双击所需的数据对象，则把该对象名称自动输入表达式一栏内；
- 设置有关的属性；
- 按“检查”按钮，进行正确性检查。检查通过后，按“确认”按钮，完成动画连接。

以四路数字量输入显示框图为例，双击 1 路数字量输入显示框图，打开单元属性设置对话框，如图 8-60 所示，在对话框的动画连接标签页下列出了可用的连接。



图8-60开关量输入的动画组态属性设置窗口

按以上步骤即可完成其余几路开关量输入的组态属性设置。

(5) 设备连接

MCGS 组态软件提供了大量的工控领域常用的设备驱动程序，同时也提供了 OPC 服务器的数据接口。通常情况下，在启动MCGS 组态软件时，模拟设备都会自动装载到设备工具箱中。如果未被装载，可按照以下步骤将其选入：

在工作台“设备窗口”中双击“设备窗口”图标进入。点击工具条中的“工具箱”图标，打开“设备工具箱”。单击“设备工具箱”中的“设备管理”按钮，弹出如图 8-61所示窗口：



图8-61 MCGS 设备管理对话框图

双击OPC 设备图标，即可将“OPC 设备”添加到右侧选定设备列表中。选中选定设备列表中的“OPC 设备”，单击“确认”，“OPC 设备”即被添加到“设备工具箱”中。OPC 设备被装载完成后，可以在MCGS 软件环境中按以下步骤添加OPC 设备，并对其属性进行设置：双击“设备工具箱”中的“OPC 设备”，OPC 设备被添加到设备组态窗口中。如图 8-62所示。



图8-62 MCGS OPC 设备添加窗口

双击“设备0-[OPC 设备]”，进入OPC 设备属性设置窗口，如图8-63所示。



图8-63 MCGS OPC 设备属性设置窗口


点击基本属性页中的“OPC 服务器”选项，该项右侧会出现图标，单击此按钮浏览计算机中可用的OPC 服务器，如图8-64所示。



图8-64 MCGS 浏览可用的OPC 服务器窗口

选中ZLGCAN OPC SERVER V2.10，单击“确定”，完成“OPC 服务器”设置。从“数据采集方式”选项的下拉列表中选择“0—同步采集”。从“初始工作状态”选项的下拉列表中选择“1—启动”。将最小采集周期改为100。点击通道连接标签，进入通道连接设置。如图8-65所示，把设置的变量与ZOPC中的通道进行连接。



图8-65 MCGS 通道连接设置对话框

进入“设备调试”属性页，即可看到通道值中数据在变化。按“确认”按钮，完成设备属性设置。

(6) 组态运行

打开文件菜单，进入运行环境，运行界面如图8-66所示，可以实现对测控实验系统的基本监控。

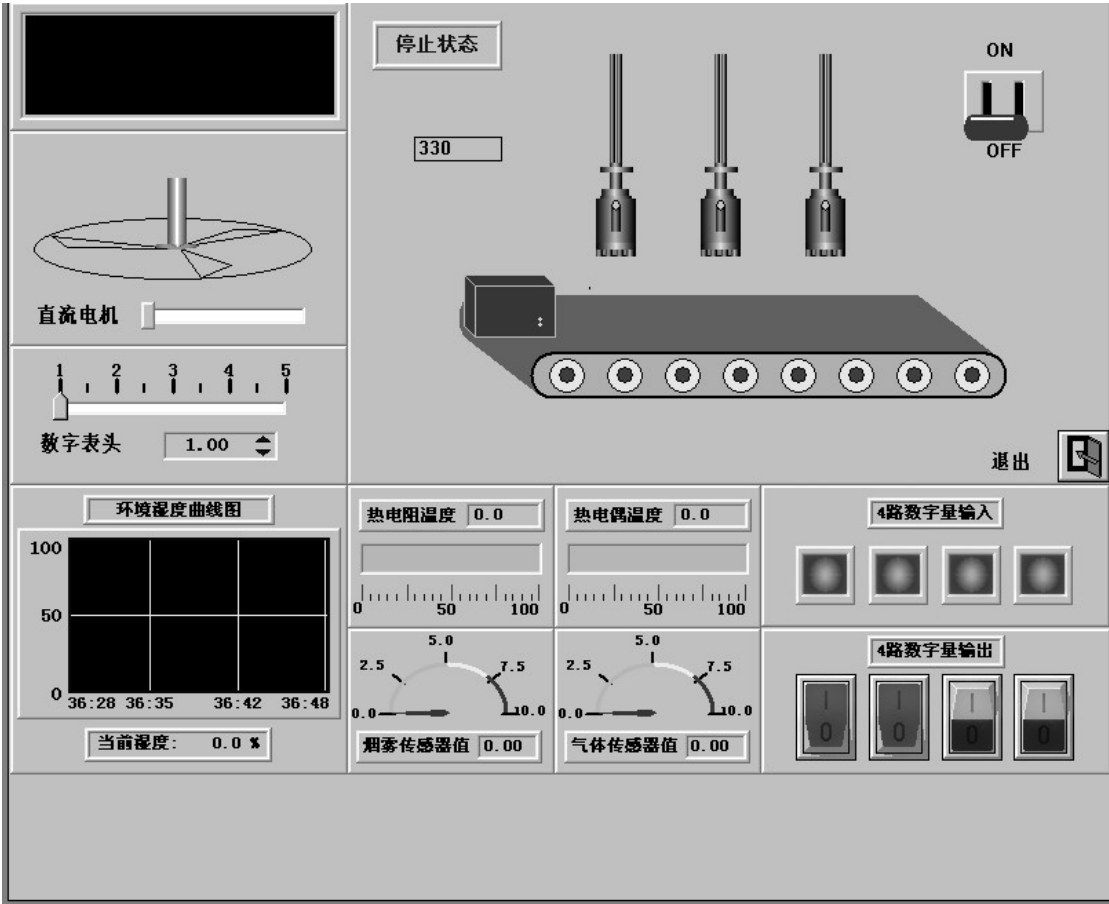


图8-66 运行界面