

中华人民共和国国家标准

GB/T 38687—2020

橡胶塑料机械 外围设备通信协议

Rubber and plastics machinery—Communicating protocol for auxiliary equipment

2020-03-31 发布

2021-02-01 实施

国家市场监督管理总局
国家标准化管理委员会 发布

目 次

前言	Ⅲ
1 范围	1
2 规范性引用文件	1
3 术语、定义和缩略语	1
4 CAN/CANopen 通信协议总体描述	2
4.1 概述	2
4.2 物理层	2
4.3 CANopen 设备结构	3
4.4 CANopen 通信协议	5
4.5 CANopen 设备子协议	7
4.6 厂商自定义子协议	8
5 CANopen 设备通信地址定义	9
5.1 概述	9
5.2 LSS Master 设备	10
5.3 LSS Slave 接口	10
5.4 外围辅助设备地址规范	10
6 CANopen 外围辅助设备自定义子协议	11
6.1 概述	11
6.2 干燥机	11
6.3 模温机	14
6.4 冷水机	16
6.5 上料机	19
7 Modbus 协议总体描述	21
7.1 概述	21
7.2 总体描述	22
8 Modbus 通信地址与通信变量定义	28
8.1 外围辅助设备地址规范	28
8.2 数据地址规范	29

前 言

本标准按照 GB/T 1.1—2009 给出的规则起草。

本标准由中国石油和化学工业联合会提出。

本标准由全国橡胶塑料机械标准化技术委员会(SAC/TC 71)归口。

本标准起草单位:东莞信易电热机械有限公司、博创智能装备股份有限公司、海天塑机集团有限公司、广东伊之密精密注压科技有限公司、桂林橡胶机械有限公司、东华机械有限公司、宁波力劲机械有限公司、泰瑞机器股份有限公司、余姚华泰橡塑机械有限公司、北京橡胶工业研究设计院有限公司、中国塑料机械工业协会。

本标准主要起草人:吴峻睿、王立斌、李崇德、焦晓龙、蒋小军、车海峰、严厚明、周刚、魏建鸿、葛福炯、何成、李春燕、谢仲铭、王更新。

橡胶塑料机械 外围设备通信协议

1 范围

本标准规定了橡胶塑料机械外围设备通信协议的背景概要、CAN/CANopen 通信协议总体描述、CANopen 设备通信地址定义、CANopen 外围辅助设备自定义子协议、Modbus 协议总体描述、Modbus 通信地址与通信变量定义。

本标准适用于橡胶塑料机械通用的成型主机设备(如注射成型机、挤出机等)与辅助主机完成加工工艺的外围辅助设备(如干燥机、模温机、冷水机、上料机等)间进行数据交换用的通信协议。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件,仅注日期的版本适用于本文件。凡是不注日期的引用文件,其最新版本(包括所有的修改单)适用于本文件。

GB/T 19582.1 基于 Modbus 协议的工业自动化网络规范 第 1 部分:Modbus 应用协议

GB/T 19582.2—2008 基于 Modbus 协议的工业自动化网络规范 第 2 部分:Modbus 协议在串行链路上的实现指南

GB/T 36587 橡胶塑料机械 术语

ISO 11898-2 道路车辆 控制器局域网(CAN) 第 2 部分:高速介质访问单元[Road vehicles—Controller area network (CAN)—Part 2:High-speed medium access unit]

3 术语、定义和缩略语

3.1 术语和定义

GB/T 19582.1、GB/T 19582.2—2008 和 GB/T 36587 界定的以及下列术语和定义适用于本文件。

3.1.1

外围辅助设备 auxiliary device;AU_D

辅助主机完成加工工艺的设备(如干燥机、模温机、冷水机、上料机等)。

3.1.2

主机设备 master device;MA_D

橡胶塑料机械通用的成型设备(如注射成型机、挤出机等)。

3.1.3

底层设置服务 layer setting services;LSS

提供查询和改变 CANopen 模块底层参数的功能。

3.1.4

层管理 layer management;LMT

查询和改变参数设置的一种 CAL 模式。

3.1.5

网络管理 network management;NMT

在 CAN 网络中用于配置、初始化和处理网络错误的应用层服务。

3.2 缩略语

下列缩略语适用于本文件。

CAN:控制器局域网(controller area network)

CiA:自动化控制器局域网(CAN in automation)

CAL:控制器局域网应用层(CAN application layer)

COB:控制器局域网对象(CAN object)

COB-ID:控制器局域网对象标识(CAN object identifier)

4 CAN/CANopen 通信协议总体描述

4.1 概述

从 OSI 网络模型的角度来看,CAN 总线只定义了 OSI 网络模型的第一层(物理层)和第二层(数据链路层),而在实际设计中,这两层完全由硬件实现,设计人员无需再为此开发相关软件或固件。

同时,CAN 只定义物理层和数据链路层,没有规定应用层,本身并不完整,因此需要一个高层协议来定义 CAN 报文中的 11 位或 29 位标识符和 8 字节数据的使用。而且,基于 CAN 总线的工业自动化应用中,越来越需要一个开放的、标准化的高层协议;这个协议支持各种 CAN 厂商设备的互用性、互换性,能够实现在 CAN 网络中提供标准的、统一的系统通信模式,提供设备功能描述方式,执行网络管理功能。

CANopen 协议是 CAN-in-Automation(CiA)定义的协议之一,并且在发布后不久就获得了广泛的承认,尤其是在欧洲,CANopen 协议被认为是在基于 CAN 的工业系统中占领导地位的协议。大多数重要的设备类型,例如数字和模拟的输入输出模块、驱动设备、操作设备、控制器、可编程控制器或编码器,都在称为“设备描述”的协议中进行描述;“设备描述”定义了不同类型的标准设备及其相应的功能。依靠 CANopen 协议的支持,可以对不同厂商的设备通过总线进行配置。

4.2 物理层

4.2.1 推荐使用两个 9 针 D-SUB 连接器,根据第二版 CiA 102 定义了连接器的各针脚(含针连接器与孔连接器)。

4.2.2 采用 CAN ISO 高速通信标准(ISO 11898-2)。

4.2.3 pin(针脚)定义:

——(pin 2)CAN 低电平;

——(pin 3)CAN 接地(仅连接于绝缘侧);

——(pin 5)CAN 屏蔽(不能连接于 9 针 D-SUB 连接器的外壳或设备外壳);

——(pin 7)CAN 高电平;

——(pin 9)可选 CAN 24 V DC。

24 V DC 电源对于 CAN 网络是可选项。

4.2.4 通信线定义:

——双绞屏蔽线,线径不小于 0.34 mm^2 ;

——CAN 低电平与 CAN 高电平(针脚 2 与针脚 7)使用一组双绞线;

——CAN 接地与 CAN 24 V DC(针脚 3 与针脚 9)使用一组双绞线;

——屏蔽线接到针脚 5,但不能连接到连接器的外壳;

——屏蔽线仅可通过主机的接地线接地。

4.2.5 所有外围辅助设备及主机设备须配备一个集成 $120\ \Omega$ 电阻的孔连接器,该连接器可以使用链条、绳索或其他类似方法固定于设备上。CAN 网络所有设备通过连接器连接。

如果已由主机控制端集成了 $120\ \Omega$ 的总线终端电阻,则设备不需要外部端接连接器。

4.2.6 主节点(主机设备)配备一个针连接器。

4.2.7 波特率为 $250\ \text{kbit/s}$ 。可以支持其他波特率。

4.2.8 应对对地绝缘电压不低于 $500\ \text{V}$ 直流电的每个节点进行直流去耦。使用波特率为 $250\ \text{kbit/s}$ 时,最小线路长度为 $180\ \text{m}$ 可保证所有节点是隔离的。

4.2.9 通信帧使用标准帧格式,不使用扩展帧格式。

4.2.10 CANopen 设备 ID 号(两位数:01~99)可在设备上通过拨码器或是参数设定等带有记忆功能的设定实现。

4.3 CANopen 设备结构

4.3.1 概述

CAN 设备以总线形式连接,通常配备两个 CAN 连接器。配线一般从主机设备的第一个连接器接出到第一台外围辅助设备的其中一个接口,由第一台外围辅助设备的另一个连接器接出再接入第二台外围辅助设备的其中一个连接器,依次类推。CAN 总线应有终止措施,避免反射。

图 1 描述了接线的基本原则。

CANopen 是一个基于 CAN 串行总线系统和 CAL(CAN 应用层)的高层协议。CANopen 的核心概念是设备对象字典(OD;ObjectDictionary),CANopen 通信通过对象字典(OD)能够访问驱动器的所有参数。

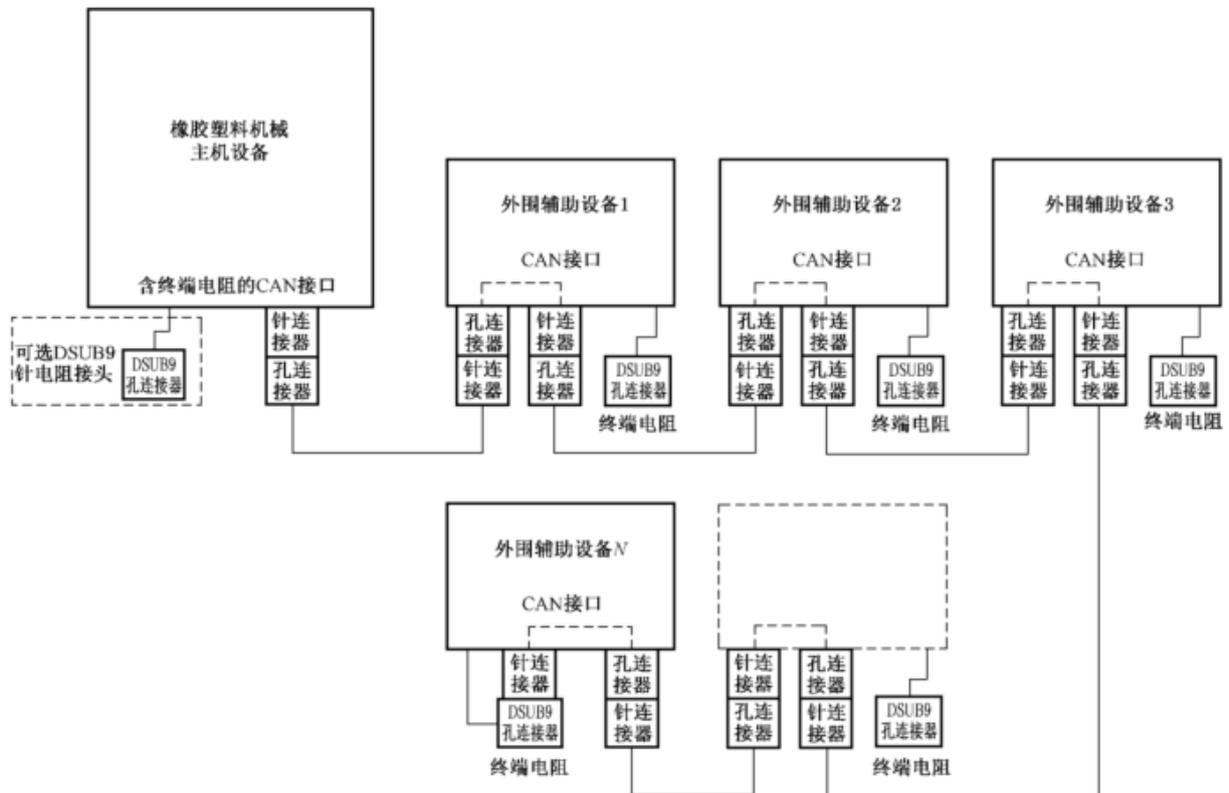


图 1 CAN 总线接线示意图

4.3.2 CANopen 对象字典

CANopen 对象字典(Object Dictionary, OD)是 CANopen 协议最为核心的概念。所谓的“对象字典”,就是一个有序的对象组;每个对象采用一个 16 位的索引值来寻址。为了访问数据结构中的元素,同时定义了一个 8 位的子索引,对象字典的结构如表 1 所示。

表 1

索引	对象
0	不使用
0001h~001Fh	静态数据类型
0020h~003Fh	复杂数据类型
0040h~005Fh	制造商规定的复杂数据类型
0060h~007Fh	设备子协议规定的静态数据类型
0080h~009Fh	设备子协议规定的复杂数据类型
00A0h~00FFh	保留
1000h~1FFFh	通信子协议区域(如设备类型、错误寄存器、支持的 PDO 数量)
2000h~5FFFh	制造商特定子协议区域
6000h~9FFFh	标准设备子协议区域
A000h~BFFFh	标准接口子协议区域
C000h~FFFFh	保留

CANopen 网络中每个节点都有一个对象字典。对象字典包含了描述这个设备和它的网络行为的所有参数。

CANopen 对象字典中的项由一系列子协议来描述。子协议描述对象字典中每个对象的功能、名字、索引、子索引、数据类型、读/写属性,以及这个对象是否必需等,从而保证不同厂商的同类型设备兼容。

CANopen 协议的核心描述子协议是 DS301,包括 CANopen 协议应用层及通信结构描述,其他子协议都是对 DS301 协议描述文本的补充与扩展。

4.3.3 CANopen 子协议

4.3.3.1 通信子协议

通信子协议(Communication Profile)描述对象字典的主要形式,以及对象字典中的通信对象和参数。这个子协议适用于所有的 CANopen 设备,其索引值范围为 1000 h~1FFFh。

4.3.3.2 制造商自定义子协议

对于在设备子协议中未定义的特殊功能,制造商可以在制造商自定义子协议(Manufacturer-specific Profile)中根据需求定义对象字典项。因此,这个区域对不同厂商来说,相同的对象字典项的定义不一定相同,其索引值范围为 2000 h~5FFFh。

4.3.3.3 设备子协议

设备子协议(Device Profile)为各种不同类型设备定义对象字典中的对象,其索引值范围为

6000 h~9FFFh。

4.4 CANopen 通信协议

4.4.1 概述

CANopen 协议是基于 CAN 总线网络。它定义了消息标识符如何被分配到设备及消息如何构成。

CANopen 设备通过节点号(节点 ID)进行识别,一般该识别号在设备中设定,例如可通过拨码器实现。

每一个 CANopen 设备的中央控制器就是一个对象字典。该目录包含设备的所有参数。这些参数由通信参数、应用参数及应用数据组成。其中部分对象在通信子协议及设备子协议中定义。此外,仍有很多空间保留为生产商扩展使用。

数据对象通常是 8 位、16 位或 32 位有符号或无符号格式。也有可能是更大的数据,如字符串。在这种情况下,需要多个 CAN 消息进行存取。

不同的服务可以建立连接、配置、同步及对象目录的常用存取。

在 CANopen 协议中主要定义网络管理对象(NMT)、服务数据对象(SDO)、过程数据对象(PDO)、预定义报文或特殊功能对象 4 种对象。

4.4.2 主机设备与 CANopen

在橡胶塑料机械行业,外围辅助设备一般指模温机、冷水机、干燥机、上料机 etc 辅助设备。主机设备即成型机,包括注塑机、挤出机等。外围辅助设备一般可由主机设备进行控制,如主机设备可以对模温机进行启停及设置温度等操作,模温机控制器也可将实际温度或其他参数值传回主机设备。

用于测量装置及闭环控制器的 CANopen 设备子协议 CiA DP-404 已经对这样的设备进行了定义。该协议支持多通道设备。

已经有大量设备制造商的设备配备了 CANopen 接口。因而,基于 CANopen 构建一个新的标准适用于成型周边辅机设备通信是非常必要的。

4.4.3 外围辅助设备接口

外围辅助设备接口基于 CANopen 协议。在此协议内,针对各种不同装置定义了不同的子协议。这些子协议定义了不同的功能块,这些功能块可用于不同的场合。

外围辅助设备定义如何使用常用的功能块。此外,CANopen 协议为设备生产商指定扩展提供了空间。它们用于在实例中描述额外的需求信息。

4.4.4 网络管理对象

网络管理对象负责层管理、网络管理和 ID 分配服务,例如,初始化、配置和网络管理。网络管理中,同一个网络中只允许有一个主节点、一个或多个从节点,并遵循主/从模式。

4.4.5 服务数据对象

服务数据对象主要用于主节点对从节点的参数配置。服务确认是 SDO 最大的特点,为每个消息都生成一个应答,以确保数据传输的准确性。在一个 CANopen 系统中,通常 CANopen 从节点作为 SDO 服务器,CANopen 主节点作为客户端。客户端通过索引和子索引能够访问数据服务器上的对象字典,所以 CANopen 主节点可以访问从节点的任意对象字典项的参数,并且 SDO 可以传输任何长度的数据。

4.4.6 过程数据对象

过程数据对象用来传输实时数据,其传输模型为生产者-消费者模型,数据长度被限制为 1~8 字节。

PDO 通信对象具有下列特点:

- a) PDO 通信没有协议规定,PDO 数据内容由它的 COB-ID 定义。
- b) 每个 PDO 在对象字典中用 2 个对象描述,分别是 PDO 通信参数和映射参数:
 - 1) PDO 通信参数:定义该设备所使用的 COB-ID、传输类型、定时周期。
 - 2) PDO 映射参数:包含一个对象字典中的对象列表,这些对象映射到相应的 PDO,其中包括数据的长度。对于生产者和消费者,只有知道这个映射参数,才能够正确地解释 PDO 的内容。PDO 内容是预定义的,如果 PDO 支持可变 PDO 映射,那么可以通过 SDO 进行配置。
- c) PDO 具有 2 种传输方式:同步传输和异步传输:
 - 1) 同步传输:通过接收同步对象实现同步,按触发方式又可分为非周期传输和周期传输。非周期传输由远程帧预触发,或者由设备子协议中规定的对象特定事件预触发。周期传输则通过接收同步对象来实现,可以设置 1~240 个同步对象触发。
 - 2) 异步传输:由特定事件触发。按触发方式又可分为 2 种:一种通过发送与 PDO 的 COB-ID 相同的远程帧来触发;另一种由设备子协议中规定的对象特定事件来触发(如定时传输、数据变化传输等)。

4.4.7 预定义通信对象

4.4.7.1 预定义

下列通信对象被预定义:

- a) 同步对象(SYNC object):
 - 1) 从站可以通过主站与同步对象同步。同步包含通信及设备中的活动两部分。
 - 2) 同步对象由主站周期性地传送给所有从站。
 - 3) 同步对象包含极高优先级的报文。报文没有数据字节。为了响应“同步”,从站可以传送合适配置的 PDO(主从原则)并且启动动作,如测量等。
- b) 紧急对象(Emergency object):参考异常处理。

4.4.7.2 异常处理

异常处理对象包含节点安全及紧急。

节点安全:节点安全可以使主站与从站相互监视。主站需要周期性地通过 RTR(远程传送)从各从站请求节点安全报文。主站可以通过“连接是否超时”监控识别节点是否异常。如果请求缺失,从站可识别异常并且按预定的行为做出反应。

紧急:从站可以通过紧急对象报告异常状态。该消息拥有高于 PDO 消息的优先级。

4.4.7.3 网络管理

CANopen 定义了简单的网络管理机制(NMT)。通过 NMT 命令可以切换不同的状态。这些状态仅在开始时由主站传送。NMT 消息拥有最高有优先级。

从站应至少支持表 2 的状态(最小设备能力)。

表 2

状态	描述
初始化	通电之后的状态。从站设备自动对其本身进行初始化并且转换为预运行状态
预运行	从站设备允许从其对象目录中以 SDO 的方式访问。这就意味着该从站可以被参数化。该状态下禁用 PDO 通信
运行	从站设备可运行。如果适宜,可进行同步,且可建立 SDO 及 PDO 通信
准备	在该状态下,禁用 SDO 及 PDO 通信,但是启用节点安全

4.4.7.4 标识符分配

消息标识符是由功能码与节点 ID 组成,表 3 为概览。预定义主/从连接集中的广播对象见表 3。

表 3

对象	功能码(二进制)	通信对象 ID(COB-ID)	备注
NMT	0000	0	主站大于所有从站
SYNC	0001	128	主站大于所有从站
时间标识	0010	256	主站大于所有从站

预定义主/从连接集中的对等网络对象见表 4。

表 4

对象	功能码(二进制)	通信对象 ID(COB-ID)	备注
紧急	0001	129~255	从站大于主站
PDO(tx)	0011	385~511	从站大于主站
PDO(rx)	0100	513~639	主站大于从站
PDO(tx)	0101	641~767	从站大于主站
PDO(rx)	0110	769~895	主站大于从站
SDO(tx)	1011	1 409~1 535	从站大于主站
SDO(rx)	1100	1 537~1 663	主站大于从站
节点安全	1110	1 793~1 919	主站大于从站

4.5 CANopen 设备子协议

4.5.1 CiA 404

当通过不同传感器或控制器获取过程数据时,可以使用 CANopen 子协议 CiA 404。基于此目的,CiA 404 定义了不同的功能块,见表 5。每个功能块最高包含 199 个通道。

表 5

功能块	应用
数字输入功能块	数字输入,每个通道可以映射 8 个数字输入
模拟输入功能块	对于传感器,该功能块可以用于构建 CANopen 接口,包括诸如传感器类型信息和单位信息
数字输出功能块	数字输出,每个通道可以映射 8 个数字输出
模拟输出功能块	模拟输出
控制器功能块	为所有类型的控制器描述各种控制器 PID 算法
报警功能块	传感器和极限值监控
设备功能块	基于此功能块产生警报;例如,传感器异常时,数据超出某个临界值或两个值的差异超过预设值

在对象目录中子协议指定的数据既可以根据功能块划分也可以根据数据类型分组,见表 6、表 7。

表 6

索引	数据类型
6000h…6FFFh	浮点数
7000h…7FFFh	16 位整数
8000h…8FFFh	24 位整数
9000h…9FFFh	32 位整数

表 7

对象类型	对象
数字输入功能块	6000h…60FFh
模拟输入功能块	X100h…X1FFh
数字输出功能块	6200h…62FFh
模拟输出功能块	X300h…X3FFh
控制器功能块	X400h…X4FFh
报警功能块	X500h…X5FFh

4.5.2 数据格式和单位

在 CiA 404 中,任何值都可声明单位。并且可以将非整型值转化为 16 位整数。从而定义在某个对象中小数点后面可以包含位数(见 CiA 404 对象 6102h)。

示例: 67.8 转化成 678,小数点后面是 1 位数。

4.6 厂商自定义子协议

4.6.1 CiA 404 对象

厂商自定义子协议使用下列 CiA 404 功能块:

- a) 控制器功能块(Controller function block);
- b) 报警功能块(Alarm function block);
- c) 本设备特有功能块。

控制器功能块包括通用控制器参数和控制方式。

报警功能块用于监控超限(下限或上限)的过程值。

4.6.2 扩展对象

原则上讲,温度控制设备反馈的控制参数被映射在由设备子协议定义的控制器功能块中。然而,当这些预定参数不能满足实际设备的需求,就需要将更多参数定义在对象目录的厂商自定义子协议部分。

采用与 CiA 404 相同的通道机制:分索引用于识别通道。如果仅 1 个通道可用,则分索引为 1。分区与子协议指定区域的一样,见表 8。

表 8

索引	数据类型
2000h...2FFFh	浮点数(不使用)
3000h...3FFFh	整数 16
4000h...4FFFh	整数 24(不使用)
5000h...5FFFh	整数 32(不使用)
注:本标准中 CANopen 自定义子协议的所有参数都定义为 16 位整数。	

过程值(如温度等)包括下列参数:

- a) 有效值;
- b) 物理单位字段;
- c) 小数位字段。

控制字包括一些扩展命令,例如,负压模式(在系统有泄漏时的一种紧急操作模式)或故障重置。各种情况下的设备动作及异常通过字状态通知主机。

该子协议仅定义了少量主要由用户预测的特定异常位。

如果一个辅机设备的异常没有相应的预定义异常位,则可使用“通用异常位”。已出现异常的详细信息可以在设备上查看。

5 CANopen 设备通信地址定义

5.1 概述

底层设置服务(LSS)和协议提供查询和修改 CANopen 设备物理层、数据链路层和应用层参数的功能。LSS 能够查询和修改的参数包括:

- a) CANopen 从节点的地址(Node-ID);
- b) 物理层的位定时参数(波特率);
- c) LSS 地址(参考 DS301 索引 1 018 h)。

通过 LSS 及协议,LSS Slave 可通过网络配置相应的参数(设置节点地址和波特率等),而无需使用硬件,如 DIP 开关等设置地址和波特率。

LSS 功能占用两个接口。在支持 LSS 的 CANopen 网络中只能有一个带有 LSS Master 的节点。LSS Master 节点通过配置 CAN 节点底层参数上的 LSS Slave 实现。

LSS Master 和 LSS Slave 之间通信遵守 LSS 协议。

5.2 LSS Master 设备

通过 CAN 网络管理其他节点模块的节点功能模块叫 LSS Master。网络上只能有一个 LSS Master。且 LSS Master 没有属性。LSS Master 具有 NMT 主能力。

5.3 LSS Slave 接口

5.3.1 概述

受 LSS Master 管理的就叫 LSS Slave。带有 LSS Slave 的节点数没有限制。LSS Slave 有下列属性。

5.3.2 LSS 地址

一个 LSS Slave 由一个 LSS 地址标识。一个 LSS 地址包括一个制造商 ID(vendor-id), 一个产品码(product-code), 一个版本修订码(revision-number) 和一个序列号(serial-number) 组成。制造商 ID 和产品码都是数字码。修订号包括数字的较大和较小修订号。序列号也是数字的。它们有下列语法:

⟨LSS-ADDRESS⟩ ::= ⟨vendor-id⟩⟨product-code⟩⟨revision-number⟩⟨serial-number⟩

⟨vendor-id⟩ ::= ‘UNSIGNED32’

⟨product-code⟩ ::= ‘UNSIGNED32’

⟨revision-number⟩ ::= ‘UNSIGNED32’

⟨serial-number⟩ ::= ‘UNSIGNED32’

对于 LSS-Address 要注意下列几点:

- LSS 地址由 CANopen 身份对象(1018h)标识;
- LSS Slave 的 LSS 地址可以被查询;
- 世界上不可能有相同⟨LSS-Address⟩的 LSS Slave 存在。

请注意: 仅当一个 LSS Slave 设备与 LSS Master 设备相连接时, 并不要求该 LSS Slave 有一个独一无二的 LSS address。

5.3.3 Node-ID

Node-ID 在 01h~7Fh 范围有效。

5.3.4 LSS 模式

激活 LSS 功能需要将所有设备的状态置为停止状态。LSS Master 应和 NMT-Master 在一个节点上。LSS 有配置和可操作两个状态。在配置状态中, 所有 LSS 服务可用, 而在可操作状态只有状态切换服务可用。

5.4 外围辅助设备地址规范

本标准适用的常用外围辅助设备有: 模温机、干燥机、冷水机及上料机。对于以上设备, 其 LSS 地址规范如表 9 所示。

表 9

外围辅助设备	供应商标识符	产品代码	版本号	系列号
模温机	保留	00000001h	厂家自定义	厂家自定义
干燥机	保留	00000002h	厂家自定义	厂家自定义
冷水机	保留	00000003h	厂家自定义	厂家自定义
上料机	保留	00000004h	厂家自定义	厂家自定义

6 CANopen 外围辅助设备自定义子协议

6.1 概述

外围辅助设备需要其他的参数值,如流量及压力等。这些参数并没有包含在标准子协议中,因而应在生产商特定区域进行描述。

为了使每个生产商的设备可能用到的参数相同,本协议定义了橡胶塑料机械外围设备的自定义子协议。

这些子协议包含了一系列的额外设备对象。然而,这并不意味着每台设备都需要使用所有对象,它仅保证所有设备指定的对象可以被各不同生产厂商以相同方法进行存取。

本协议存在于不同类型的设备中。本协议并非全新协议,而是基于 CiA 404(用于测量装置及闭环控制器的 CANopen 设备子协议)定义了额外的对象目录,以期适用于橡胶塑料机械的辅机设备。

6.2 干燥机

6.2.1 综述

基于 CiA 404(用于测量装置及闭环控制器的 CANopen 设备子协议)定义了额外的对象目录,以期适用于干燥机。

6.2.2 扩展对象

以下对象(见表 10~表 15)应在对象目录中的厂商特定区域实现。

设备和 CANopen 接口应建立一个区域。没有在设备中实现的对象不应在 CANopen 接口实现。

注:本标准中,温度数值的小数点后保留 1 位。

示例:543 代表 54.3 ℃,也就是说,分辨率为 1/10 ℃。以下各表都沿用该单位。

表 10

索引	名称	数据类型	单位	备注	读写类型	
2000h	设备配置文件	STRING	—		只读	
2001h	机器状态	UNSIGNED 16	—	0:状态无效;1:状态有效	只读	
				bit0 待机状态		bit8 保留
				bit1 运行状态		bit9 保留
				bit2 延时停机状态		bit10 保留

表 10 (续)

索引	名称	数据类型	单位	备注				读写类型
				bit3	故障状态	bit11	保留	
2001h	机器状态	UNSIGNED 16	—	bit4	保留	bit12	保留	只读
				bit5	保留	bit13	保留	
				bit6	保留	bit14	保留	
				bit7	保留	bit15	保留	
				bit3	故障状态	bit11	保留	
2002h	开关量输入状态	UNSIGNED 16	—	0:正常;1:输入异常				只读
				bit0	风机过载输入	bit8	保留	
				bit1	超温输入	bit9	保留	
				bit2	保留	bit10	保留	
				bit3	保留	bit11	保留	
				bit4	保留	bit12	保留	
				bit5	保留	bit13	保留	
				bit6	保留	bit14	保留	
				bit7	保留	bit15	保留	
2003h	继电器输出状态	UNSIGNED 16	—	0:关闭;1:开启				只读
				bit0	风机输出	bit8	保留	
				bit1	电热输出	bit9	保留	
				bit2	脱扣输出	bit10	保留	
				bit3	报警输出	bit11	保留	
				bit4	保留	bit12	保留	
				bit5	保留	bit13	保留	
				bit6	保留	bit14	保留	
				bit7	保留	bit15	保留	
2004h	故障信息	UNSIGNED 16	—	0:无故障;1:有故障				只读
				bit0	风机过载报警	bit8	保留	
				bit1	超温保护器报警	bit9	保留	
				bit2	超温报警	bit10	保留	
				bit3	温度传感器异常	bit11	保留	
				bit4	加热器失效报警	bit12	保留	
				bit5	温度过低报警	bit13	保留	
				bit6	保留	bit14	保留	
				bit7	保留	bit15	保留	
2005h	保留	UNSIGNED 16	—	—				只读

表 10 (续)

索引	名称	数据类型	单位	备注	读写类型
2010h	温度实际值	INTEGER 16	℃或℉	干燥机的温度实际值	只读
2011h	设备累积运行时间	INTEGER 16	小时(h)	记录设备累积运行时间	只读
2012h	保留	INTEGER 16	—	—	只读
2013h	保留	INTEGER 16	—	—	只读
2014h	温度设定值	INTEGER 16	℃或℉	干燥机的温度设定值	读/写
2015h	开关机	INTEGER 16	—	0:干燥机关机; 1:干燥机开机	读/写
2016h	最高温度	INTEGER 16	℃或℉	定义干燥机的最高工作温度	读/写
2017h	温度单位	INTEGER 16	—	0:温度单位为摄氏度(℃); 1:温度单位为华氏度(℉)	读/写
2018h	保留	INTEGER 16	—	—	读/写
2019h	保留	INTEGER 16	—	—	读/写
2020h	保留	INTEGER 16	—	—	读/写
2021h	保留	INTEGER 32	—	—	读/写
2022h	保留	INTEGER 32	—	—	读/写
2023h	保留	INTEGER 32	—	—	读/写
2024h	保留	INTEGER 32	—	—	读/写

6.2.3 设备配置文件

索引为 2000h 的对象用于描述符合本协议设备的类型与版本。它包含一个 8 位字段(用于描述本协议的代码)及其他三个 8 位字段(用于描述设备配置文件信息)。以下模温机、冷水机、上料机此项均相同,故不再重复定义。

该对象定义了本协议的设备配置文件,见表 11、表 12。

表 11

索引	分索引	名称	类型	属性	默认值
2000h	0	设备配置文件	变量	只读	01 00 01 50h

表 12

最高有效位→最低有效位			
子协议版本	子协议索引	子协议代码	协议代码

协议代码:80(50h)。

子协议代码:干燥机:01;

模温机:02h;

冷水机:03h;

上料机:04h。

子协议版本和索引与设备的设备子协议版本一致。

例如:

文件版本 1.0:子协议版本 01h 和子协议索引 00h。

文件版本 3.4:子协议版本 03h 和子协议索引 40h。

6.3 模温机

扩展对象见表 13。

表 13

索引	名称	数据类型	单位	备注	读写类型	
2001h	机器状态	UNSIGNED 16	—	0:状态无效;1:状态有效	只读	
				bit0 待机状态		bit8 保留
				bit1 运行状态		bit9 保留
				bit2 延时停机状态		bit10 保留
				bit3 故障状态		bit11 保留
				bit4 保留		bit12 保留
				bit5 保留		bit13 保留
				bit6 保留		bit14 保留
				bit7 保留		bit15 保留
2002h	开关量输入状态 1	UNSIGNED 16	—	0:正常;1:输入异常	只读	
				bit0 泵过载输入		bit8 保留
				bit1 超温输入		bit9 保留
				bit2 低液位输入		bit10 保留
				bit3 高液位输入		bit11 保留
				bit4 压力过低输入		bit12 保留
				bit5 压力过高输入		bit13 保留
				bit6 保留		bit14 保留
				bit7 保留		bit15 保留
2003h	开关量输入状态 2	UNSIGNED 16	—	0:正常;1:输入异常	只读	
				bit0 保留		bit8 保留
				bit1 保留		bit9 保留
				bit2 保留		bit10 保留
				bit3 保留		bit11 保留
				bit4 保留		bit12 保留
				bit5 保留		bit13 保留
				bit6 保留		bit14 保留
				bit7 保留		bit15 保留

表 13 (续)

索引	名称	数据类型	单位	备注	读写类型			
2004h	继电器输出状态	UNSIGNED 16	—	0:关闭;1:开启	只读			
				bit0		泵正转	bit8	保留
				bit1		泵反转	bit9	保留
				bit2		加热输出	bit10	保留
				bit3		冷却输出	bit11	保留
				bit4		脱扣输出	bit12	保留
				bit5		报警输出	bit13	保留
				bit6		保留	bit14	保留
				bit7		保留	bit15	保留
2005h	故障信息 1	UNSIGNED 16	—	0:无故障;1:有故障	只读			
				bit0		泵过载报警	bit8	加热器失效报警
				bit1		超温报警	bit9	温度干扰报警
				bit2		压力过低报警	bit10	温度过低报警
				bit3		压力过高报警	bit11	保留
				bit4		低液位报警	bit12	保留
				bit5		相位报警	bit13	保留
				bit6		温度传感器异常	bit14	保留
				bit7		温度偏差报警	bit15	保留
2006h	故障信息 2	UNSIGNED 16	—	0:无故障;1:有故障	只读			
				bit0		保留	bit8	保留
				bit1		保留	bit9	保留
				bit2		保留	bit10	保留
				bit3		保留	bit11	保留
				bit4		保留	bit12	保留
				bit5		保留	bit13	保留
				bit6		保留	bit14	保留
				bit7		保留	bit15	保留
2007h	保留	UNSIGNED 16	—	—	只读			
2008h	保留	UNSIGNED 16	—	—	只读			
2009h	保留	UNSIGNED 16	—	—	只读			

表 13 (续)

索引	名称	数据类型	单位	备注	读写类型
2010h	控制温度实际值	INTEGER 16	℃或℉	模温机的控制温度实际值	只读
2011h	温度测量值 1	INTEGER 16	℃或℉	模温机附加的温度测量值 1(如模具温度值)	只读
2012h	温度测量值 2	INTEGER 16	℃或℉	模温机附加的温度测量值 2(如回液温度值)	只读
2013h	流量	INTEGER 16	升每分 (L/min)	模温机的实际流量	只读
2014h	压力	INTEGER 16	巴(bar)	模温机的实际压力(1 bar=10 ⁵ Pa)	只读
2015h	温控调节比例	INTEGER 16	%	实际温控调节比例(加热为正数,冷却为负数)	只读
2016h	设备累积运行时间	INTEGER 32	小时(h)	记录设备累积运行时间	只读
2017h	保留	INTEGER 16	—	—	只读
2018h	保留	INTEGER 16	—	—	只读
2019h	保留	INTEGER 16	—	—	只读
2020h	控制温度设定值	INTEGER 16	℃或℉	模温机的控制温度设定值	读/写
2021h	开关机	UNSIGNED 8	—	0:模温机关机;1:模温机开机	读/写
2022h	最高温度	INTEGER 16	℃或℉	定义模温机的最高工作温度	读/写
2023h	温度单位	UNSIGNED 8	—	0:温度单位为摄氏度(℃);1:温度单位为华氏度(℉)	读/写
2024h	保留	INTEGER 32	—	—	读/写
2025h	保留	INTEGER 32	—	—	读/写
2026h	保留	INTEGER 32	—	—	读/写
2027h	保留	INTEGER 32	—	—	读/写
2028h	保留	INTEGER 32	—	—	读/写
2029h	保留	INTEGER 32	—	—	读/写
2030h	保留	INTEGER 32	—	—	读/写

6.4 冷水机

扩展对象见表 14。

表 14

索引	名称	数据类型	单位	备注	读写类型	
2001h	机器状态	UNSIGNED 16	—	0:状态无效;1:状态有效	只读	
				bit0 待机状态		bit8 保留
				bit1 运行状态		bit9 保留
				bit2 延时停机状态		bit10 保留
				bit3 故障状态		bit11 保留

表 14 (续)

索引	名称	数据类型	单位	备注				读写类型
2001h	机器状态	UNSIGNED 16	—	bit4	保留	bit12	保留	只读
				bit5	保留	bit13	保留	
				bit6	保留	bit14	保留	
				bit7	保留	bit15	保留	
2002h	开关量输入状态 1	UNSIGNED 16	—	0:正常;1:输入异常				只读
				bit0	远程输入	bit8	防冻开关	
				bit1	压缩机 1 高压	bit9	三相电源开关	
				bit2	压缩机 1 低压	bit10	压缩机 2 高压	
				bit3	压缩机 1 过载	bit11	压缩机 2 低压	
				bit4	循环泵过载	bit12	压缩机 2 过载	
				bit5	风机 1 过载	bit13	风机 2 过载	
				bit6	水位开关	bit14	保留	
2003h	开关量输入状态 2	UNSIGNED 16	—	0:正常;1:输入异常				只读
				bit0	保留	bit8	保留	
				bit1	保留	bit9	保留	
				bit2	保留	bit10	保留	
				bit3	保留	bit11	保留	
				bit4	保留	bit12	保留	
				bit5	保留	bit13	保留	
				bit6	保留	bit14	保留	
2004h	继电器输出状态	UNSIGNED 16	—	0:关闭;1:开启				只读
				bit0	循环泵	bit8	保留	
				bit1	风机 1	bit9	保留	
				bit2	压缩机 1	bit10	保留	
				bit3	旁通阀	bit11	保留	
				bit4	报警	bit12	保留	
				bit5	风机 2	bit13	保留	
				bit6	压缩机 2	bit14	保留	
bit7	保留	bit15	保留					

表 14 (续)

索引	名称	数据类型	单位	备注	读写类型			
2005h	故障信息 1	UNSIGNED 16	—	0:无故障;1:有故障		只读		
				bit0	压缩机 1 高压		bit8	电源故障
				bit1	压缩机 1 低压		bit9	液温温度过低
				bit2	压缩机 1 过载		bit10	液温温度过高
				bit3	水位故障		bit11	液温探头断路
				bit4	水流故障		bit12	液温探头短路
				bit5	循环泵过载		bit13	参数异常
				bit6	风机 1 过载		bit14	压缩机运行超时
				bit7	防冻故障		bit15	环境温度过高
2006h	故障信息 2	UNSIGNED 16	—	0:无故障;1:有故障		只读		
				bit0	环温探头断路		bit8	风机 2 过载
				bit1	环温探头短路		bit9	保留
				bit2	防冻探头断路		bit10	保留
				bit3	防冻探头短路		bit11	保留
				bit4	防冻温度过低		bit12	保留
				bit5	压缩机 2 高压		bit13	保留
				bit6	压缩机 2 低压		bit14	保留
				bit7	压缩机 2 过载		bit15	保留
2007h	保留	UNSIGNED 16	—	—	只读			
2008h	保留	UNSIGNED 16	—	—	只读			
2009h	保留	UNSIGNED 16	—	—	只读			
2010h	温度实际值	INTEGER 16	℃或℉	冷水机出水温度的实际值	只读			
2011h	防冻温度	INTEGER 16	℃或℉	冷水机的防冻温度	只读			
2012h	环境温度	INTEGER 16	℃或℉	冷水机的环境温度	只读			
2013h	流量	INTEGER 16	升每分 (L/min)	冷水机的实际流量	只读			
2014h	压力	INTEGER 16	巴(bar)	冷水机出水的实际压力(1 bar=10 ⁵ Pa)	只读			
2015h	设备累积运行时间	INTEGER 32	小时(h)	记录设备累积运行时间	只读			
2016h	保留	INTEGER 16	—	—	只读			

表 14 (续)

索引	名称	数据类型	单位	备注	读写类型
2017h	保留	INTEGER 16	—	—	只读
2018h	保留	INTEGER 16	—	—	只读
2019h	保留	INTEGER 16	—	—	只读
2020h	温度设定值	INTEGER 16	℃或℉	冷水机的温度设定值	读/写
2021h	水泵启停开关	UNSIGNED 8	—	0:水泵停机; 1:水泵启动	读/写
2022h	压缩机启停开关	UNSIGNED 8	—	0:压缩机停机; 1:压缩机启动	读/写
2023h	温度单位	UNSIGNED 8	—	0:温度单位为摄氏度(℃); 1:温度单位为华氏度(℉)	读/写
2024h	上限温度	INTEGER 16	℃或℉	定义了冷水机的上限温度	读/写
2025h	下限温度	INTEGER 16	℃或℉	定义了冷水机的下限温度	读/写
2026h	保留	INTEGER 16	—	—	读/写
2027h	保留	INTEGER 16	—	—	读/写
2028h	保留	INTEGER 32	—	—	读/写
2029h	保留	INTEGER 32	—	—	读/写
2030h	保留	INTEGER 32	—	—	读/写

6.5 上料机

扩展对象见表 15。

表 15

索引	名称	数据类型	单位	备注	读写类型	
2001h	机器状态	UNSIGNED 16	—	0:状态无效;1:状态有效	只读	
				bit0 待机状态		bit8 保留
				bit1 吸料状态		bit9 保留
				bit2 清洗状态		bit10 保留
				bit3 故障状态		bit11 保留
				bit4 保留		bit12 保留
				bit5 保留		bit13 保留
				bit6 保留		bit14 保留
				bit7 保留		bit15 保留

表 15 (续)

索引	名称	数据类型	单位	备注	读写类型	
2002h	开关量输入状态	UNSIGNED 16	—	0:正常;1:输入异常	只读	
				bit0 电机过载输入		bit8 保留
				bit1 缺料输入		bit9 保留
				bit2 保留		bit10 保留
				bit3 保留		bit11 保留
				bit4 保留		bit12 保留
				bit5 保留		bit13 保留
				bit6 保留		bit14 保留
2003h	继电器输出状态	UNSIGNED 16	—	0:关闭;1:开启	只读	
				bit0 电机输出		bit8 保留
				bit1 清洗输出		bit9 保留
				bit2 混料输出		bit10 保留
				bit3 截料输出		bit11 保留
				bit4 破真空阀输出		bit12 保留
				bit5 报警输出		bit13 保留
				bit6 保留		bit14 保留
2004h	故障信息 1	UNSIGNED 16	—	0:无故障;1:有故障	只读	
				bit0 电机过载报警		bit8 保留
				bit1 滤网阻塞报警		bit9 保留
				bit2 缺料报警		bit10 保留
				bit3 缺料停机报警		bit11 保留
				bit4 保留		bit12 保留
				bit5 保留		bit13 保留
				bit6 保留		bit14 保留
2005h	保留	UNSIGNED 16	—	—	只读	
2010h	设备累积运行时间	UNSIGNED 16	小时(h)	记录设备累积运行时间	只读	
2011h	保留	UNSIGNED 16	—	—	只读	
2012h	保留	UNSIGNED 16	—	—	只读	
2013h	保留	UNSIGNED 16	—	—	只读	
2014h	吸料时间设定值	UNSIGNED 16	秒(s)	上料机的吸料时间设定值	读/写	

表 15 (续)

索引	名称	数据类型	单位	备注	读写类型
2015h	开关机	UNSIGNED 8	—	0:上料机关机; 1:上料机开机	读/写
2016h	清网间隔	UNSIGNED 16	次	上料机按照一定间隔清网的次数	读/写
2017h	清洗时间	UNSIGNED 16	秒(s)	上料机实际清洗所需要的时间	读/写
2018h	缺料报警次数	UNSIGNED 16	次	上料机工作过程中几次缺料后才输出报警	读/写
2019h	保留	UNSIGNED 16	—	—	读/写
2020h	保留	UNSIGNED 16	—	—	读/写
2021h	保留	UNSIGNED 16	—	—	读/写
2022h	保留	UNSIGNED 32	—	—	读/写
2023h	保留	UNSIGNED 32	—	—	读/写
2024h	保留	UNSIGNED 32	—	—	读/写

7 Modbus 协议总体描述

7.1 概述

Modbus 是 OSI 模型第 7 层上的应用层报文传输协议(见图 2),它在连接至不同类型总线或网络的设备之间提供客户机/服务器通信。

目前,通过下列方式实现 Modbus 通信:

- 以太网上的 TCP/IP;
- 各种介质(有线:EIA/TIA-232-E、EIA-422、EIA/TIA-485-A;光纤、无线等)上的异步串行传输;
- Modbus+,一种高速令牌传递网络。

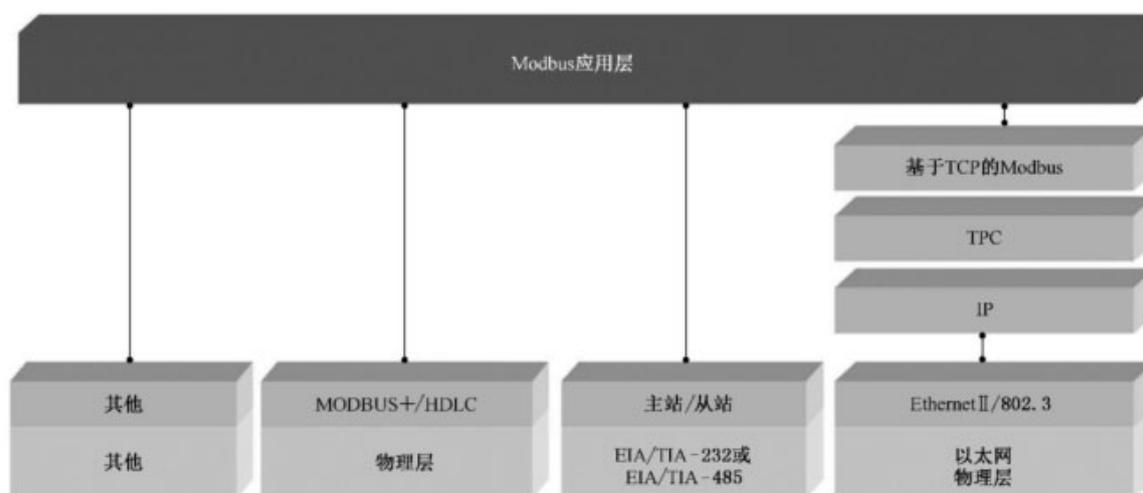


图 2 Modbus 通信栈

7.2 总体描述

7.2.1 协议描述

Modbus 协议定义了一个与基础通信层无关的简单协议数据单元(PDU)。特定总线或网络上的 Modbus 协议映射能够在应用数据单元(ADU)上引入一些附加字段,见图 3。

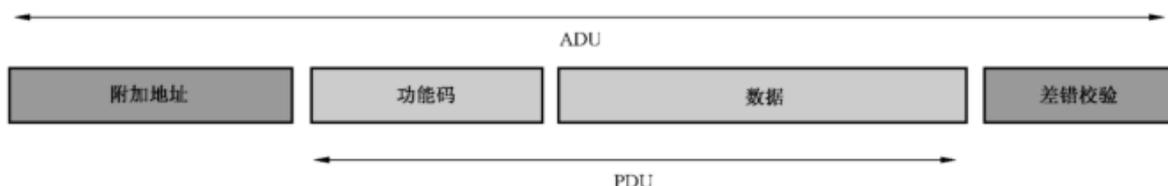


图 3 通用 Modbus 帧

Modbus 应用数据单元由启动 Modbus 事务处理的客户机创建。功能码向服务器指示将执行哪种操作。Modbus 应用协议建立了客户机启动的请求格式。

用一个字节编码 Modbus 数据单元的功能码字段。有效的码范围是十进制 1~255(128~255 保留用于异常响应)。当从客户机向服务器设备发送报文时,功能码字段通知服务器执行哪种操作。功能码“0”无效。可以向一些功能码加入子功能码来定义多项操作。

当服务器对客户机响应时,它使用功能码字段来指示正常(无差错)响应(见图 4)或者出现某种差错(称为异常响应),见图 5。对于正常响应,服务器仅复制原始功能码。

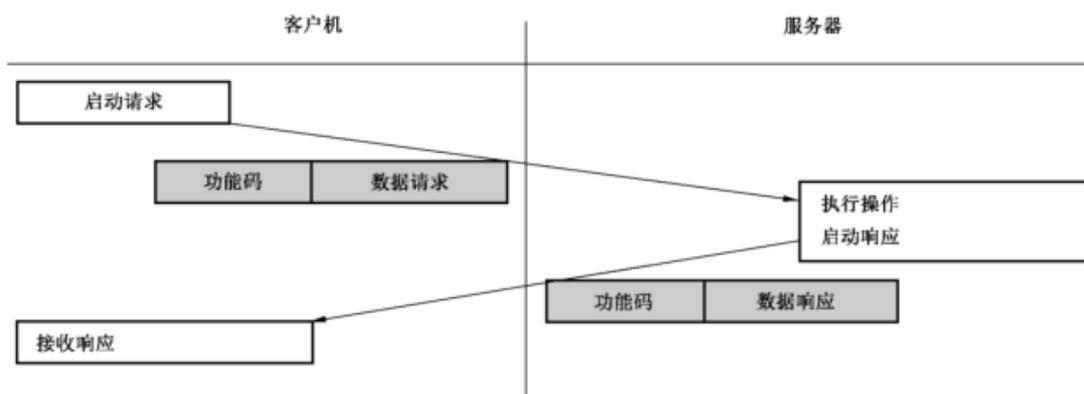


图 4 Modbus 事务处理(无差错)

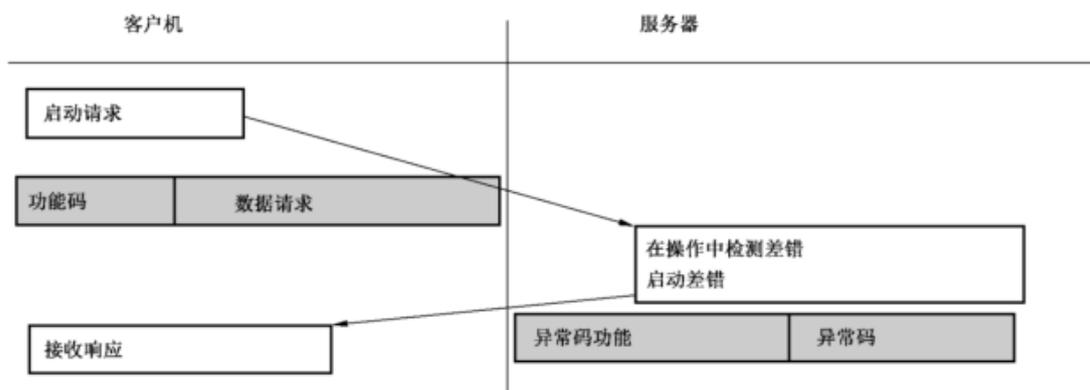


图 5 Modbus 事务处理(异常响应)

Modbus 最初在串行链路上的实现(最大 RS485 ADU=256 字节)限制了 Modbus PDU 的长度。因此,对串行链路通信来说,Modbus PDU=256-服务器地址(1 字节)-CRC(2 字节)=253 字节。

从而:

RS232/RS485 ADU=253 字节+服务器地址(1 字节)+CRC(2 字节)=256 字节。

Modbus 协议定义了三种 PDU。它们是:

- Modbus 请求 PDU,mb_req_pdu;
- Modbus 响应 PDU,mb_rsp_pdu;
- Modbus 异常响应 PDU,mb_except_rsp_pdu。

7.2.2 物理层

7.2.2.1 物理接口

在物理层,串行链路上的 Modbus 系统可以使用不同的物理接口(RS485、RS232)。最常用的物理接口是 EIA/TIA-485(也称 RS485 标准)二线制接口。由于 RS232 存在通信距离近等缺点,本标准采用最常用的 EIA/TIA-485(也称 RS485 标准)二线制接口。

7.2.2.2 2 线 Modbus 定义

Modbus 在串行链路上的解决方案应该依照 EIA/TIA-485 标准实现“2 线”电气接口。

在这个 2 线总线上,在任何时候只有一个驱动器有权发送信号。

实际上,还应使用第三条导线将总线上所有设备相互连接:公共端。

图 6 显示了通用 2 线拓扑结构。

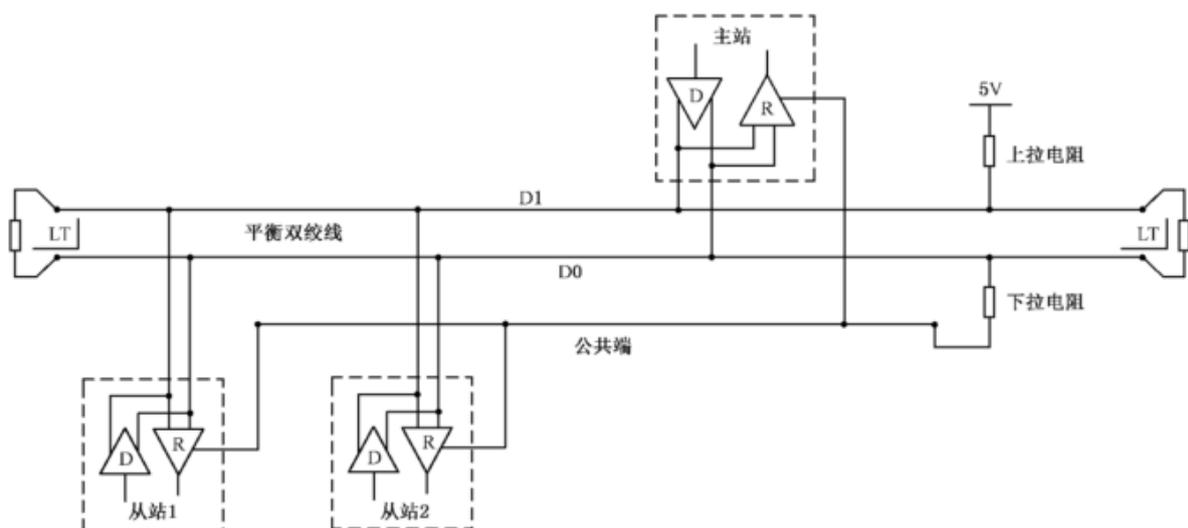


图 6 通用的 2 线拓扑结构

表 16 则说明 2 线 Modbus 电路定义。

表 16

在电路上要求的		设备	在设备上要求的	EIA/TIA-485 的名称	描述
在 ITr 上	在 IDv 上				
D1	D1	I/O	X	B/B'	收发器端子 1, U_1 电压 ($U_1 > U_0$, 对于二进制 1[OFF]状态)
D0	D0	I/O	X	A/A'	收发器端子 0, U_0 电压 ($U_0 > U_1$, 对于二进制 0[ON]状态)
公共端	公共端	—	X	C/C'	信号和可选电源的公共端

Modbus 在串行链路上的解决方案应该依照 EIA/TIA-485 标准实现“2 线”电气接口。

在这个 2 线总线上,在任何时候只有一个驱动器有权发送信号。

实际上,还应使用第三条导线将总线上所有设备的公共端相互连接。

7.2.2.3 EIA/TIA-485 多点系统要求

7.2.2.3.1 无中继器情况下,最大设备数量

在没有中继器的 RS485 Modbus 系统中,最多允许有 32 个设备。

与下列项目有关:

- 所有可能的地址;
- 设备使用的 RS485 单元负载总量;
- 以及需要的线路极性偏置。

一个 RS485 系统可以容纳许多设备。有些设备在没有中继器情况下允许在设备数大于 32 个的 RS485-Modbus 串行链路上运行。

在这种情况下,应在这个 Modbus 设备文件中说明没有中继器时能允许接这类设备的数量。

也可以在两个重负载的 RS485 Modbus 之间使用中继器。

7.2.2.3.2 拓扑结构

没有配置中继器的 RS485 Modbus 有一个与所有设备直接连接(菊花链)或通过短分支电缆连接的干线电缆。

干线电缆,又称总线,可能很长。它的两端应接线路终端。

也可以在多个 RS485 Modbus 之间使用中继器。

7.2.2.3.3 长度

应限制干线电缆的端到端长度。最大长度与波特率、电缆(规格、电容或特性阻抗)、菊花链上的负载数量以及网络配置(2 线制或 4 线制)有关。

对于最高波特率为 9 600 bit/s、AWG26(或更粗)规格的电缆来说,其最大长度为 100 m。

分支应短,不能超过 20 m。如果使用 n 个分支的多端口分支器,每个分支最大长度应限制为 40 m 除以 n 。

7.2.2.3.4 接地形式

应将“公共端”电路(信号与可选电源的公共端)直接连接到保护地上,最好是整条总线单点接地。

通常,该点可选在主站上或其分支器上。

7.2.2.3.5 线路终端

沿线路传播的信号遇到阻抗不连续,会在传输线路中产生反射。为了使从 RS485 电缆端的反射最小,要求在总线接近两端处放置线路终端。

由于传播是双向的,故在线路两端配置终端是非常重要的,但是,在一个无源 D0-D1 平衡线对上放置的线路终端不允许超过 2 个。也不允许在分支电缆上放置任何线路终端。

每个线路终端应连接在平衡线 D0 和 D1 的两条导线之间。

线路终端可以是 $150\ \Omega$ (0.5 W) 的电阻。

当双绞线应进行极性偏置时,较好的选择是使用电容 (1 nF, 最低 10 V) 与 $120\ \Omega$ (0.25 W) 电阻串联。

7.2.2.4 机械接口

图 7 为 2 线 Modbus 中使用的 RJ45 连接器(要求的插脚引线)。

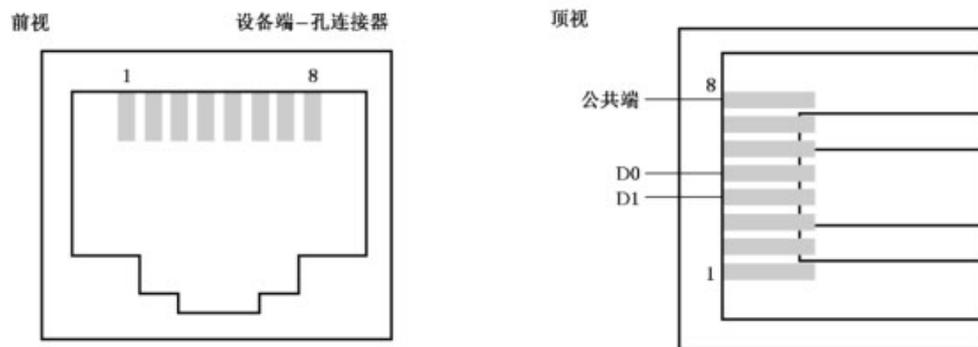


图 7 RJ45 连接器

图 8 为 9 针 D-SUB 连接器。

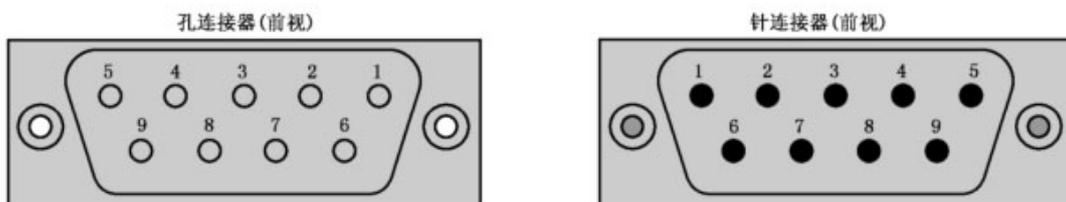


图 8 9 针 D-SUB 连接器

2 线 Modbus RJ45 和 9 针 D-SUB 连接器引脚分配如表 17 所示。

表 17

RJ45 引脚	D9-型 连接器引脚	要求的 等级	IDv 电路	ITr 电路	EIT/TIA-485 名称	IDv 的描述
3	3	可选的	PMC	—	—	端口模式控制
4	5	要求的	D1	D1	B/B'	收发器端子 1, U_1 电压 ($U_1 > U_0$, 对于二进制的 1[OFF]状态)

表 17 (续)

RJ45 引脚	D9-型 连接器引脚	要求的 等级	IDv 电路	ITr 电路	EIT/TIA-485 名称	IDv 的描述
5	9	要求的	D0	D0	A/A'	收发器端子 O, U_0 电压 ($U_0 > U_1$, 对于二进制的 0[ON]状态)
7	2	建议的	VP	—	—	正的 5…24VDC 电源
8	1	建议的	公共端	公共端	C/C'	信号和电源的公共端

7.2.3 数据编码

Modbus 使用最高有效字节在低地址存储的方式表示地址和数据项。这意味着当发送多个字节时,首先发送最高有效字节。

示例:

在发送 16 位数据“1234h”时,首先发送的第一字节“12h”,然后再发送“34h”。

7.2.4 Modbus 数据模型

Modbus 的数据模型是以一组具有不同特征的表为基础建立的。4 个基本表见表 18。

表 18

基本表	对象类型	访问类型	注释
离散量输入	单个位	只读	I/O 系统可提供这种类型的数据
线圈	单个位	读写	通过应用程序可改变这种类型的数据
输入寄存器	16 位字	只读	I/O 系统可提供这种类型的数据
保持寄存器	16 位字	读写	通过应用程序可改变这种类型的数据

7.2.5 Modbus 串行链路协议

7.2.5.1 概述

Modbus 串行链路协议是一个主从协议。在同一时间,总线上只能有一个主站和一个或多个(最多 247 个)从站。Modbus 通信总是由主站发起。当从站没有收到来自主站的请求时,不会发送数据。从站之间不能相互通信。主站同时只能启动一个 Modbus 事务处理。

主站用两种模式向从站发出 Modbus 请求:

- 单播模式,主站寻址单个从站。从站接收并处理完请求之后,向主站返回一个报文(一个“应答”)。在这种模式下,一个 Modbus 事务处理包含 2 个报文:一个是主站的请求,另一个是从站的应答。每个从站应有唯一的地址(1~247),这样才能区别于其他站独立地被寻址;
- 广播模式,主站可以向所有的从站发送请求。对于主站发送的广播请求没有应答返回。广播请求应是写命令。所有设备应接受广播方式的写命令。地址 0 被保留用来识别广播通信。

7.2.5.2 Modbus 寻址规则

Modbus 寻址空间由 256 个不同地址组成。见表 19。

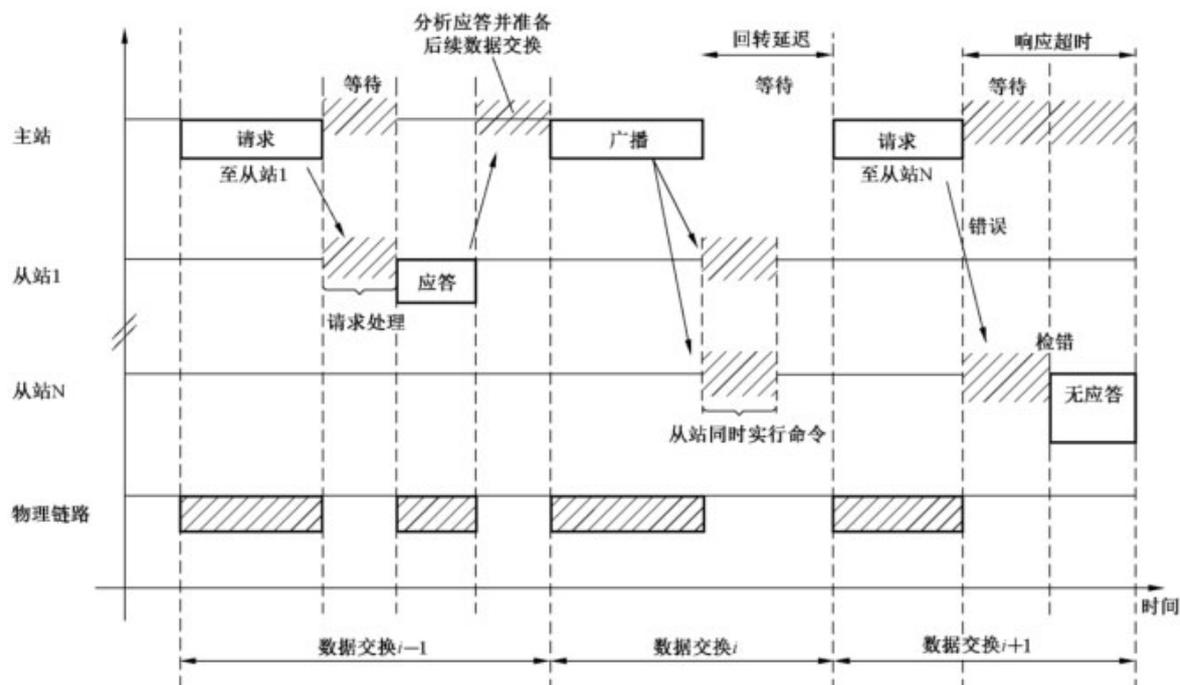
地址 0 为广播地址。所有从站应识别广播地址。

表 19

寻址空间	0	1~247	248~255
描述	广播地址	从站地址	保留

7.2.5.3 主站/从站通信时序图

主站/从站通信时序图见图 9。



注 1: 请求、应答、广播阶段的持续时间与通信特征(帧长度和吞吐量)有关。

注 2: 等待和处理阶段的持续时间与从站应用所需的请求处理时间有关。

图 9 主站/从站通信时序图

7.2.5.4 两种串行传输模式

7.2.5.4.1 概述

定义了两种串行传输模式:RTU 模式和 ASC II 模式。

定义了链路上串行传送报文的位内容。它确定了信息如何打包为报文和如何解码。

在 Modbus 串行链路上,所有设备的传输模式(及串行口参数)应相同。

用户应该将设备设置成所期望的模式:RTU 或 ASC II 模式。默认设置应为 RTU 模式。

7.2.5.4.2 RTU 传输模式

当设备在 Modbus 串行链路上使用 RTU(远程终端单元)模式通信时,报文中每个 8 位字节含有两个 4 位十六进制字符。这种模式的主要优点是具有较高的字符密度,在相同的波特率下,比 ASC II 模式有更高的数据吞吐量。每个报文应以连续的字符流传输。

RTU 模式中每个字节(11 位)的格式为:

- 编码系统:8 位二进制;
- 每个字节的位:1 个起始位;
 - 8 个数据位,首先发送最低有效位;
 - 1 个奇偶校验位;
 - 1 个停止位。

要求使用偶校验。也可以使用其他模式(奇校验、无校验)。为了保证与其他产品的最大兼容性,建议还支持无校验模式。默认校验模式应是偶校验。

注:使用无校验时要求 2 个停止位。

RTU 报文帧及 RTU 传输模式状态图,详见 GB/T 19582.2—2008 中 6.5.1。

7.2.5.4.3 ASC II 传输模式

当使用 ASC II(美国信息交换标准代码)模式设置设备在 Modbus 串行链路上通信时,用两个 ASC II 字符发送报文中的一个 8 位字节。当物理通信链路或者设备能力不能满足 RTU 模式的定时管理要求时,使用该模式。

注 1:由于每个字节需要两个字符发送,所以这种模式比 RTU 模式效率低。

示例:将字节 5Bh 编码分为两个字符:35 h 和 42 h(用 ASC II 表示的 35 h="5",42 h="B")。

ASC II 模式中每个字节的格式为(10 位):

编码系统:十六进制,ASC II 字符 0~9、A~F;

在报文中每个 ASC II 字符中,1 个十六进制字符包含 4 个数据位。

- 每个字节的位:1 个起始位;
 - 7 个数据位,首先发送最低有效位;
 - 1 个奇偶校验位;
 - 1 个停止位。

要求使用偶校验。也可以使用其他模式(奇校验、无校验)。为了保证与其他产品的最大兼容性,建议还支持无校验模式。默认校验模式应是偶校验。

注 2:使用无校验时要求 2 个停止位。

ASC II 报文帧及 ASC II 传输模式状态图,详见 GB/T 19582.2—2008 中 6.5.2。

7.2.5.4.4 数据信号传输速率

要求实现 9 600 bit/s 和 19.2 kbit/s 传输速率。默认值 19.2 kbit/s。

也可使用其他波特率:1 200 bit/s、2 400 bit/s、4 800 bit/s……38 400 bit/s、56 kbit/s、115 kbit/s 等。

在发送的情况下,每种实现的波特率精度应高于 1%;在接收的情况下,应允许 2%误差。

所有设备的传输速率应相同。

8 Modbus 通信地址与通信变量定义

8.1 外围辅助设备地址规范

外围辅助设备地址规范见表 20。

表 20

外围辅助设备	地址
模温机	1…10
干燥机	11…20

表 20 (续)

外围辅助设备	地址
冷水机	21…30
上料机	31…40

8.2 数据地址规范

8.2.1 干燥机

8.2.1.1 输入寄存器

干燥机输入寄存器见表 21。使用功能码 04h 从干燥机设备中读输入寄存器。请求 PDU 指定了起始地址和寄存器数量。编号为 1~16 的输入寄存器被寻址为 0~15。

表 21

输入寄存器地址 (30000+i)(十进制)	数据描述	单位	备注 [功能码 04h(读输入寄存器)]			
30001 (i=1)	机器状态	—	0:状态无效;1:状态有效			
			bit0	待机状态	bit8	保留
			bit1	运行状态	bit9	保留
			bit2	延时停机状态	bit10	保留
			bit3	故障状态	bit11	保留
			bit4	保留	bit12	保留
			bit5	保留	bit13	保留
			bit6	保留	bit14	保留
30002 (i=2)	开关量输入状态	—	0:正常;1:输入异常			
			bit0	风机过载输入	bit8	保留
			bit1	超温输入	bit9	保留
			bit2	保留	bit10	保留
			bit3	保留	bit11	保留
			bit4	保留	bit12	保留
			bit5	保留	bit13	保留
			bit6	保留	bit14	保留
30003 (i=3)	继电器输出状态	—	0:关闭;1:开启			
			bit0	风机输出	bit8	保留
			bit1	电热输出	bit9	保留
			bit2	脱扣输出	bit10	保留

表 21 (续)

输入寄存器地址 (30000+i)(十进制)	数据描述	单位	备注 [功能码 04h(读输入寄存器)]			
30003 (i=3)	继电器输出状态	—	bit3	报警输出	bit11	保留
			bit4	保留	bit12	保留
			bit5	保留	bit13	保留
			bit6	保留	bit14	保留
			bit7	保留	bit15	保留
30004 (i=4)	故障信息	—	0:无故障,1:有故障			
			bit0	风机过载报警	bit8	保留
			bit1	超温保护器报警	bit9	保留
			bit2	超温报警	bit10	保留
			bit3	温度传感器异常	bit11	保留
			bit4	加热器失效报警	bit12	保留
			bit5	温度过低报警	bit13	保留
			bit6	保留	bit14	保留
bit7	保留	bit15	保留			
30005 (i=5)	保留	—	—			
30006 (i=6)	温度实际值	℃或℉	干燥机的温度实际值			
30007 (i=7)	设备累积运行时间	小时(h)	记录设备累积运行时间			
30008 (i=8)	保留	—	—			
30009 (i=9)	保留	—	—			
注:i表示偏移地址。						

8.2.1.2 保持寄存器

干燥机保持寄存器见表 22。使用功能码 03h 从干燥机中读保持寄存器连续块的内容。请求 PDU 指定了起始寄存器地址和寄存器数量。在 PDU 中,从零开始寻址寄存器。因此编号 1~16 的寄存器被寻址为 0~15。将响应报文中的寄存器数据按每个寄存器两字节进行打包,这个二进制内容正好填满每个字节。对于每个寄存器,第一个字节包括高位位,第二个字节包括低位位。

使用功能码 06h 在干燥机中写单个保持寄存器。请求 PDU 指定了被写入寄存器的地址。从零开始寻址寄存器。因此,编号为 1 的寄存器被寻址为 0。正常的响应是请求的复制,在写入寄存器内容之后被返回。

使用功能码 10h 在干燥机中写连续寄存器块(1~123 个寄存器)。在请求数据字段中指定了请求写入的值。将数据按每个寄存器两字节打包。正常的响应返回功能码、起始地址以及被写入寄存器的数量。

表 22

保持寄存器地址 (40000+i)(十进制)	数据描述	单位	备注 [功能码 03h(读保持寄存器)] [功能码 06h(写单个保持寄存器)] [功能码 10h(写多个保持寄存器)]
40001 (i=1)	温度设定值	℃或℉	干燥机的温度设定值
40002 (i=2)	最高温度	℃或℉	定义干燥机的最高工作温度
40003 (i=3)	开关机	—	0:干燥机关机;1:干燥机开机
40004 (i=4)	温度单位	—	0:温度单位为摄氏度(℃); 1:温度单位为华氏度(℉)
40005 (i=5)	保留	—	—
40006 (i=6)	保留	—	—
40007 (i=7)	保留	—	—
40008 (i=8)	保留	—	—
40009 (i=9)	保留	—	—

注: i 表示偏移地址。

8.2.2 模温机

8.2.2.1 输入寄存器

模温机输入寄存器见表 23。使用功能码 04h 从模温机设备中读输入寄存器。请求 PDU 指定了起始地址和寄存器数量。

表 23

输入寄存器地址 (30000+i)(十进制)	数据描述	单位	备注 [功能码 04h(读输入寄存器)]			
30001 (i=1)	机器状态	—	0:状态无效;1:状态有效			
			bit0	待机状态	bit8	保留
			bit1	运行状态	bit9	保留
			bit2	延时停机状态	bit10	保留
			bit3	故障状态	bit11	保留
			bit4	保留	bit12	保留
			bit5	保留	bit13	保留
			bit6	保留	bit14	保留
bit7	保留	bit15	保留			

表 23 (续)

输入寄存器地址 (30000+i)(十进制)	数据描述	单位	备注 [功能码 04h(读输入寄存器)]			
30002 (i=2)	开关量输入状态 1	—	0:正常;1:输入异常			
			bit0	泵过载输入	bit8	保留
			bit1	超温输入	bit9	保留
			bit2	低液位输入	bit10	保留
			bit3	高液位输入	bit11	保留
			bit4	压力过低输入	bit12	保留
			bit5	压力过高输入	bit13	保留
			bit6	保留	bit14	保留
bit7	保留	bit15	保留			
30003 (i=3)	开关量输入状态 2	—	0:正常;1:输入异常			
			bit0	保留	bit8	保留
			bit1	保留	bit9	保留
			bit2	保留	bit10	保留
			bit3	保留	bit11	保留
			bit4	保留	bit12	保留
			bit5	保留	bit13	保留
			bit6	保留	bit14	保留
bit7	保留	bit15	保留			
30004 (i=4)	继电器输出状态	—	0:关闭;1:开启			
			bit0	泵正转	bit8	保留
			bit1	泵反转	bit9	保留
			bit2	加热输出	bit10	保留
			bit3	冷却输出	bit11	保留
			bit4	脱扣输出	bit12	保留
			bit5	报警输出	bit13	保留
			bit6	保留	bit14	保留
bit7	保留	bit15	保留			
30005 (i=5)	故障信息 1	—	0:无故障;1:有故障			
			bit0	泵过载报警	bit8	加热器失效报警
			bit1	超温报警	bit9	温度干扰报警
			bit2	压力过低报警	bit10	温度过低报警
			bit3	压力过高报警	bit11	保留
bit4	低液位报警	bit12	保留			

表 23 (续)

输入寄存器地址 (30000+i)(十进制)	数据描述	单位	备注 [功能码 04h(读输入寄存器)]			
			bit5	相位报警	bit13	保留
30005 (i=5)	故障信息 1	—	bit6	温度传感器异常	bit14	保留
			bit7	温度偏差报警	bit15	保留
			0:无故障;1:有故障			
30006 (i=6)	故障信息 2	—	bit0	保留	bit8	保留
			bit1	保留	bit9	保留
			bit2	保留	bit10	保留
			bit3	保留	bit11	保留
			bit4	保留	bit12	保留
			bit5	保留	bit13	保留
			bit6	保留	bit14	保留
			bit7	保留	bit15	保留
30007 (i=7)	保留	—	—			
30008 (i=8)	保留	—	—			
30009 (i=9)	保留	—	—			
30010 (i=10)	控制温度实际值	℃或℉	模温机的控制温度实际值			
30011 (i=11)	温度测量值 1	℃或℉	模温机附加的温度测量值 1(如模具温度值)			
30012 (i=12)	温度测量值 2	℃或℉	模温机附加的温度测量值 2(如回液温度值)			
30013 (i=13)	流量	升每分 (L/min)	模温机的实际流量			
30014 (i=14)	压力	巴(bar)	模温机的实际压力(1 bar=10 ⁵ Pa)			
30015 (i=15)	温控调节比例	%	实际温控调节比例(加热为正数,冷却为负数)			
30016 (i=16)	设备累积运行时间	小时(h)	记录设备累积运行时间			
30017 (i=17)	保留	—	—			
30018 (i=18)	保留	—	—			
30019 (i=19)	保留	—	—			

注: i 表示偏移地址。

8.2.2.2 保持寄存器

模温机保持寄存器见表 24。使用功能码 03h 从模温机中读保持寄存器连续块的内容。请求 PDU 指定了起始寄存器地址和寄存器数量。在 PDU 中,从零开始寻址寄存器。因此编号 1~16 的寄存器被寻址为 0~15。将响应报文中的寄存器数据按每个寄存器两字节进行打包,这个二进制内容正好填满每个字节。对于每个寄存器,第一个字节包括高位位,第二个字节包括低位位。

使用功能码 06h 在模温机中写单个保持寄存器。请求 PDU 指定了被写入寄存器的地址。从零开始寻址寄存器。因此,编号为 1 的寄存器被寻址为 0。正常的响应是请求的复制,在写入寄存器内容之后被返回。

使用功能码 10h 在模温机中写连续寄存器块(1~123 个寄存器)。在请求数据字段中指定了请求写入的值。将数据按每个寄存器两字节打包。正常的响应返回功能码、起始地址以及被写入寄存器的数量。

表 24

保持寄存器地址 (40000+i)(十进制)	数据描述	单位	备注 [功能码 03h(读保持寄存器)] [功能码 06h(写单个保持寄存器)] [功能码 10h(写多个保持寄存器)]
40001 (i=1)	控制温度设定值	℃或℉	模温机的控制温度设定值
40002 (i=2)	最高温度	℃或℉	定义模温机的最高工作温度
40003 (i=3)	开关机	—	0:模温机关机;1:模温机开机
40004 (i=4)	温度单位	—	0:温度单位为摄氏度(℃); 1:温度单位为华氏度(℉)
40005 (i=5)	保留	—	—
40006 (i=6)	保留	—	—
40007 (i=7)	保留	—	—
40008 (i=8)	保留	—	—
40009 (i=9)	保留	—	—
注:i表示偏移地址。			

8.2.3 冷水机

8.2.3.1 输入寄存器

冷水机输入寄存器见表 25。使用功能码 04h 从冷水机设备中读输入寄存器。请求 PDU 指定了起始地址和寄存器数量。编号为 1~16 的输入寄存器被寻址为 0~15。

表 25

输入寄存器地址 (30000+i)(十进制)	数据描述	单位	备注 [功能码 04h(读输入寄存器)]			
30001 (i=1)	机器状态	—	0:状态无效;1:状态有效			
			bit0	待机状态	bit8	保留
			bit1	运行状态	bit9	保留
			bit2	延时停机状态	bit10	保留
			bit3	故障状态	bit11	保留
			bit4	保留	bit12	保留
			bit5	保留	bit13	保留
			bit6	保留	bit14	保留
30002 (i=2)	开关量输入状态 1	—	0:正常;1:输入异常			
			bit0	远程输入	bit8	防冻开关
			bit1	压缩机 1 高压	bit9	三相电源开关
			bit2	压缩机 1 低压	bit10	压缩机 2 高压
			bit3	压缩机 1 过载	bit11	压缩机 2 低压
			bit4	循环泵过载	bit12	压缩机 2 过载
			bit5	风机 1 过载	bit13	风机 2 过载
			bit6	水位开关	bit14	保留
30003 (i=3)	开关量输入状态 2	—	0:正常;1:输入异常			
			bit0	保留	bit8	保留
			bit1	保留	bit9	保留
			bit2	保留	bit10	保留
			bit3	保留	bit11	保留
			bit4	保留	bit12	保留
			bit5	保留	bit13	保留
			bit6	保留	bit14	保留
30004 (i=4)	继电器输出状态	—	0:关闭;1:开启			
			bit0	循环泵	bit8	保留
			bit1	风机 1	bit9	保留
			bit2	压缩机 1	bit10	保留
			bit3	旁通阀	bit11	保留
bit4	报警	bit12	保留			

表 25 (续)

输入寄存器地址 (30000+i)(十进制)	数据描述	单位	备注 [功能码 04h(读输入寄存器)]			
			bit5	bit6	bit7	bit8
30004 (i=4)	继电器输出状态	—	bit5	风机 2	bit13	保留
			bit6	压缩机 2	bit14	保留
			bit7	保留	bit15	保留
30005 (i=5)	故障信息 1	—	0:无故障;1:有故障			
			bit0	压缩机 1 高压	bit8	电源故障
			bit1	压缩机 1 低压	bit9	液温温度过低
			bit2	压缩机 1 过载	bit10	液温温度过高
			bit3	水位故障	bit11	液温探头断路
			bit4	水流故障	bit12	液温探头短路
			bit5	循环泵过载	bit13	参数异常
			bit6	风机 1 过载	bit14	压缩机运行超时
30006 (i=6)	故障信息 2	—	0:无故障;1:有故障			
			bit0	环温探头断路	bit8	风机 2 过载
			bit1	环温探头短路	bit9	保留
			bit2	防冻探头断路	bit10	保留
			bit3	防冻探头短路	bit11	保留
			bit4	防冻温度过低	bit12	保留
			bit5	压缩机 2 高压	bit13	保留
			bit6	压缩机 2 低压	bit14	保留
bit7	压缩机 2 过载	bit15	保留			
30007 (i=7)	保留	—	—			
30008 (i=8)	保留	—	—			
30009 (i=9)	保留	—	—			
30010 (i=10)	温度实际值	℃或℉	冷水机出水温度的实际值			
30011 (i=11)	防冻温度	℃或℉	冷水机的防冻温度			
30012 (i=12)	环境温度	℃或℉	冷水机的环境温度			
30013 (i=13)	流量	升每分 (L/min)	冷水机的实际流量			

表 25 (续)

输入寄存器地址 (30000+i)(十进制)	数据描述	单位	备注 [功能码 04h(读输入寄存器)]
30014 (i=14)	压力	巴(bar)	冷水机出水的实际压力(1 bar=10 ⁵ Pa)
30015 (i=15)	设备累积运行时间	小时(h)	记录设备累积运行时间
30016 (i=16)	保留	—	—
30017 (i=17)	保留	—	—
30018 (i=18)	保留	—	—
30019 (i=19)	保留	—	—
注: i 表示偏移地址。			

8.2.3.2 保持寄存器

冷水机保持寄存器见表 26。使用功能码 03h 从冷水机中读保持寄存器连续块的内容。请求 PDU 指定了起始寄存器地址和寄存器数量。在 PDU 中,从零开始寻址寄存器。因此编号 1~16 的寄存器被寻址为 0~15。将响应报文中的寄存器数据按每个寄存器两字节进行打包,这个二进制内容正好填满每个字节。对于每个寄存器,第一个字节包括高位位,第二个字节包括低位位。

使用功能码 06h 在冷水机中写单个保持寄存器。请求 PDU 指定了被写入寄存器的地址。从零开始寻址寄存器。因此,编号为 1 的寄存器被寻址为 0。正常的响应是请求的复制,在写入寄存器内容之后被返回。

使用功能码 10h 在冷水机中写连续寄存器块(1~123 个寄存器)。在请求数据字段中指定了请求写入的值。将数据按每个寄存器两字节打包。正常的响应返回功能码、起始地址以及被写入寄存器的数量。

表 26

保持寄存器地址 (40000+i)(十进制)	数据描述	单位	备注 [功能码 03h(读保持寄存器)] [功能码 06h(写单个保持寄存器)] [功能码 10h(写多个保持寄存器)]
40001 (i=1)	温度设定值	℃或℉	冷水机的温度设定值
40002 (i=2)	上限温度	℃或℉	定义了冷水机的上限温度
40003 (i=3)	下限温度	℃或℉	定义了冷水机的下限温度

表 26 (续)

保持寄存器地址 (40000+i)(十进制)	数据描述	单位	备注 [功能码 03h(读保持寄存器)] [功能码 06h(写单个保持寄存器)] [功能码 10h(写多个保持寄存器)]
40004 (i=4)	水泵启停开关	—	0:水泵停机; 1:水泵启动
40005 (i=5)	压缩机启停开关	—	0:压缩机停机; 1:压缩机启动
40006 (i=6)	温度单位	—	0:温度单位为摄氏度(℃) 1:温度单位为华氏度(℉)
40007 (i=7)	保留	—	—
40008 (i=8)	保留	—	—

注: i 表示偏移地址。

8.2.4 上料机

8.2.4.1 输入寄存器

上料机输入寄存器见表 27。使用功能码 04h 从上料机设备中读输入寄存器。

表 27

输入寄存器地址 (30000+i)(十进制)	数据描述	单位	备注 [功能码 04h(读输入寄存器)]
30001 (i=1)	机器状态	—	0:状态无效; 1:状态有效
			bit0 待机状态 bit8 保留
			bit1 吸料状态 bit9 保留
			bit2 清洗状态 bit10 保留
			bit3 故障状态 bit11 保留
			bit4 保留 bit12 保留
			bit5 保留 bit13 保留
			bit6 保留 bit14 保留
30002 (i=2)	开关量输入状态	—	0:正常; 1:输入异常
			bit0 电机过载输入 bit8 保留
			bit1 缺料输入 bit9 保留
			bit2 保留 bit10 保留
			bit3 保留 bit11 保留

表 27 (续)

输入寄存器地址 (30000+i)(十进制)	数据描述	单位	备注 [功能码 04h(读输入寄存器)]			
			bit4	保留	bit12	保留
30002 (i=2)	开关量输入状态	—	bit4	保留	bit12	保留
			bit5	保留	bit13	保留
			bit6	保留	bit14	保留
			bit7	保留	bit15	保留
30003 (i=3)	继电器输出状态	—	0:关闭;1:开启			
			bit0	电机输出	bit8	保留
			bit1	清洗输出	bit9	保留
			bit2	混料输出	bit10	保留
			bit3	截料输出	bit11	保留
			bit4	破真空阀输出	bit12	保留
			bit5	报警输出	bit13	保留
			bit6	保留	bit14	保留
30004 (i=4)	故障信息 1	—	0:无故障;1:有故障			
			bit0	电机过载报警	bit8	保留
			bit1	滤网阻塞报警	bit9	保留
			bit2	缺料报警	bit10	保留
			bit3	缺料停机报警	bit11	保留
			bit4	保留	bit12	保留
			bit5	保留	bit13	保留
			bit6	保留	bit14	保留
30005 (i=5)	保留	—	—			
30006 (i=6)	设备累积运行时间	小时(h)	记录设备累积运行时间			
30007 (i=7)	保留	—	—			
30008 (i=8)	保留	—	—			
30009 (i=9)	保留	—	—			
注:i表示偏移地址。						

8.2.4.2 保持寄存器

上料机保持寄存器见表 28。使用功能码 03h 从上料机中读保持寄存器连续块的内容。请求 PDU 指定了起始寄存器地址和寄存器数量。在 PDU 中,从零开始寻址寄存器。因此编号 1~16 的寄存器被寻址为 0~15。将响应报文中的寄存器数据按每个寄存器两字节进行打包,这个二进制内容正好填满每个字节。对于每个寄存器,第一个字节包括高位位,第二个字节包括低位位。

使用功能码 06h 在上料机中写单个保持寄存器。请求 PDU 指定了被写入寄存器的地址。从零开始寻址寄存器。因此,编号为 1 的寄存器被寻址为 0。正常的响应是请求的复制,在写入寄存器内容之后被返回。

使用功能码 10h 在上料机中写连续寄存器块(1~123 个寄存器)。在请求数据字段中指定了请求写入的值。将数据按每个寄存器两字节打包。正常的响应返回功能码、起始地址以及被写入寄存器的数量。

表 28

保持寄存器地址 (40000+i)(十进制)	数据描述	单位	备注 [功能码 03h(读保持寄存器)] [功能码 06h(写单个保持寄存器)] [功能码 10h(写多个保持寄存器)]
40001 (i=1)	吸料时间设定值	秒(s)	上料机的吸料时间设定值
40002 (i=2)	清网间隔	次	上料机按照一定间隔清网的次数
40003 (i=3)	清洗时间	秒(s)	上料机实际清洗所需要的时间
40004 (i=4)	缺料报警次数	次	上料机工作过程中缺料了几次后才输出报警
40005 (i=5)	开关机	—	0:上料机关机;1:上料机开机
40006 (i=6)	保留	—	—
40007 (i=7)	保留	—	—
40008 (i=8)	保留	—	—
40009 (i=9)	保留	—	—
40010 (i=10)	保留	—	—
注:i 表示偏移地址。			