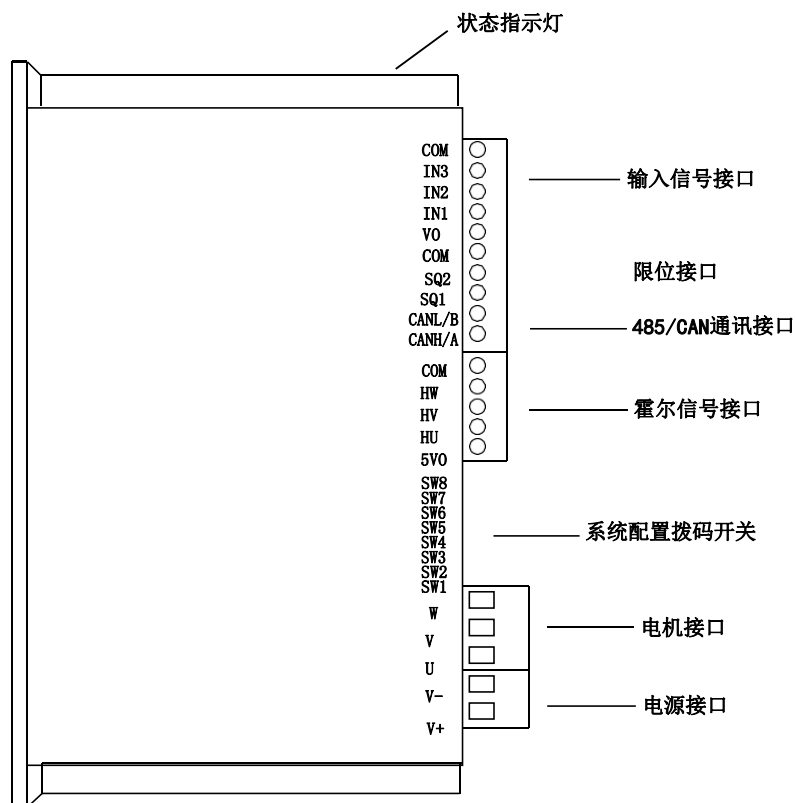


无刷电机驱动器使用手册



注意：电源接口和电机接口的接线千万不能搭在一起，它们也不能与输入信号、霍尔信号、限位或通讯接口搭在一起，否则可能损坏驱动器。电源地或控制信号的地也不要与机壳相连，否则可能造成驱动器工作不稳定。

系统配置拨码开关

在使用本驱动器前首先要对电机额定电流、信号源选择和工作模式等进行配置。通过拨码开关可以配置电机在数字/模拟信号控制方式下电机的额定电流、信号源和工作模式，以及 485/CAN 通讯控制方式下的从站地址。

通过对电机额定电流的配置，一方面设定了电机的最大负载电流，当电机过负载或堵转时，驱动器会将输出电流稳流至额定电流，有效地保护电机；另一方面可使相应额定电流的电机调速更稳定。

通过对信号源的选择，可支持用户所使用的不同的控制信号。

通过对工作模式的配置，更换相位或线序后先对电机进行相序学习才能使用；通过选择不同的调速方式可满足用户不同的应用需求；通过学习电机行程，用户可以使用模拟信号、来调节电机在固定行程内的转动位置。

系统配置拨码开关如 图 2.2 所示。开关拨到下方为ON，上方为OFF。从左至右依次是第 1~8 位。

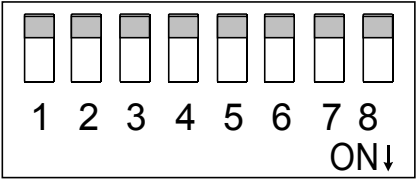


图 2.2 系统配置拨码开关

其中第 8 位为控制方式选择位。当第 8 位为 OFF 时，为模拟信号控制方式；当第 8 位为 ON 时，为 485/CAN 通讯控制方式。

注意：在使用拨码开关配置参数时，请断掉驱动器电源再进行配置，配置好后再上电。

数字/模拟信号控制方式下拨码开关各位功能定义

数字/模拟信号控制方式下拨码开关各位功能定义如 表 2.1 所示。

表 2.1 数字/模拟信号控制方式下拨码开关各位功能定义

第 1-3 位	第 4-5 位	第 6-7 位	第 8 位
电机额定电流配置	信号源的选择	工作模式的配置	控制方式位，数字/模拟信号方式请拨到 OFF

注：在数字/模拟信号控制方式下，若未指定 485/CAN 站点地址（即 0x009c 和 0x0121 寄存器配置为 0），则站点地址为 0x01；485 波特率为 9600bps，校验方式偶校验，停止位为 1 位；CAN 波特率为 500kbps。在此控制方式下，485/CAN 仅能配置参数和读取状态，不能控制电机转动。

数字/模拟信号控制方式下信号源的选择

表 2.3 数字/模拟信号控制方式下信号源选择表

第 4 位	第 5 位	第 8 位	信号源
ON	OFF	OFF	模拟信号
ON	ON	OFF	内置程序

数字/模拟信号控制方式下，信号源可选择为模拟信号或内置程序。
当信号源为模拟信号时，使用模拟信号进行调速、力矩控制或固定行程内的位置调节。
当信号源为内置程序时，工作模式可配置为电机学习、行程学习和预设速度控制方式。
详见 2.1.4 小节。

数字/模拟信号控制方式下工作模式的配置

数字/模拟信号控制方式下工作模式配置表如 表 2.4 所示。

表 2.4 数字/模拟信号控制方式下工作模式配置表

第 4 位	第 5 位	第 6 位	第 7 位	工作模式
不同时为 ON		OFF	OFF	占空比调速
		ON	OFF	力矩控制
		OFF	ON	速度闭环控制
		ON	ON	位置闭环控制
同时为 ON		OFF	OFF	电机学习
		ON	OFF	行程学习
		OFF	ON	预设速度控制
		ON	ON	保留

数字/模拟信号控制方式下，当信号源为模拟信号时，工作模式可配置为占空比、力矩、速度闭环和位置闭环控制方式。

占空比调速方式通过改变等效输出电压来调节电机转速，具有响应快的特点，但转速受负载变化有一定程度的变化，且堵转时的扭矩与占空比有关。

力矩控制方式通过调节输出电流来改变电机扭矩。力矩控制方式下支持仅力矩控制和力矩转速同时控制两种方式。仅力矩控制方式下，当负载力矩小于电机扭矩时，电机转速最终将达到最大转速。在力矩转速同时控制方式下，除了可以调节电机扭矩外，还可调节电机最终达到的转速。

速度闭环控制方式使用 PID 调节算法来对电机进行稳速控制。稳速算法支持速度闭环

控制和时间-位置闭环控制。前者直接对电机转速进行调节，具有超调量小和在高速时调速平稳的特点，但在低速时，可能出现调速不均匀问题；后者通过计算电机随时间改变应该转动的位置来对电机转动位置进行控制，从而间接对电机进行了稳速控制，此方式可满足多台驱动器对多个电机转动位置进行同步控制的要求以及超低速稳速控制的要求，但转速调节有一定超调。

位置闭环控制使用 PID 调节算法来对电机转动位置进行控制。当给定目标位置后，驱动器会根据配置的加速加速度、减速加速度和最大速度，自动计算电机运行过程中当前转动位置的目标实时速度并进行调控，从而使电机按照配置的速度和加速度参数准确地转动到目标位置。

当信号源为内置程序时，工作模式可配置为电机学习、行程学习和预设速度控制方式。

通讯控制方式下拨码开关的配置

通讯控制方式下拨码开关各位功能定义如 表 2.5 所示。

表 2.5 485/CAN 通讯控制方式下拨码开关各位功能定义

第 1-7 位	第 8 位	控制方式
全为 ON	ON	默认通讯参数控制方式
从站地址/节点 ID	ON	RS485/CAN 通讯控制方式

注：

- 1) 默认通讯参数控制方式不支持CANopen 协议；
- 2) 须 0x009c 和 0x0121 寄存器值为 0 时拨码开关配置的从站地址/节点 ID 才有效。

设备从站地址译码表如 表 2.6 所示(即二进制方式)。

表 2.6 拨码开关 1-7 位对应的从站地址译码表

第 1 位	第 2 位	第 3 位	第 4 位	第 5 位	第 6 位	第 7 位	译码值
OFF	OFF	OFF	OFF	OFF	OFF	OFF	0x01
ON	OFF	OFF	OFF	OFF	OFF	OFF	0x02
OFF	ON	OFF	OFF	OFF	OFF	OFF	0x03
ON	ON	OFF	OFF	OFF	OFF	OFF	0x04
OFF	OFF	ON	OFF	OFF	OFF	OFF	0x05
ON	OFF	ON	OFF	OFF	OFF	OFF	0x06
OFF	ON	ON	OFF	OFF	OFF	OFF	0x07
.....						
OFF	ON	ON	ON	ON	ON	ON	0x7F

电源接口

电源接口的信号定义如图 2.3 所示。V+为电源正极，V-为电源负极，电源接口支持电压范围为DC9V~60V。

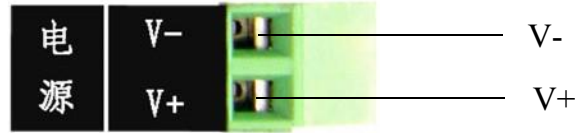


图 2.3 电源接口信号定义

电机接口

电机接口的定义如图 2.4 所示。U、V、W与电机的U、V、W相线相连(可不按顺序连接，当电机的相线顺序改变后需要重新对电机进行学习)。



图 2.4 电机接口信号定义

霍尔信号接口

霍尔信号接口定义如图 2.5 所示，COM接霍尔传感器的负极，5V0接霍尔传感器的正极，HW、HV、HU分别接霍尔传感器的三霍尔信号线(电机霍尔传感器的电源正负极一定要接正确，霍尔位置信号HW、HV、HU可不按顺序连接，当霍尔位置信号接线顺序改变后需要重新对电机进行学习)。

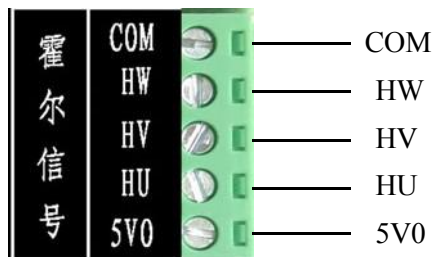


图 2.5 霍尔信号接口定义

通讯接口

485/CAN接线方法

通讯接口支持 485/CAN通讯，信号定义如图 2.6 所示。

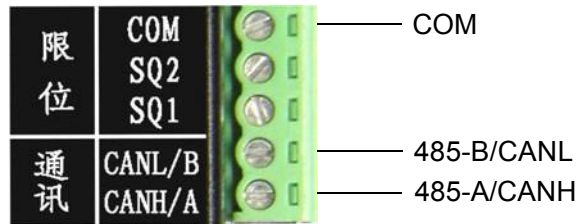


图 2.6 通讯接口信号定义

当使用 485 设备进行通讯时，485-A/CANH 和 485-B/CANL 分别为 RS485 的两差分信号 A 和 B，COM 为信号地。A 接 485 主站的信号线 A，B 接 485 主站的信号线 B。

当使用 CAN 设备进行通讯时，485-A/CANH 和 485-B/CANL 分别为 CAN 的两差分信号 CANH 和 CANL，COM 为信号地。CANH 接 CAN 总线的 CANH，CANL 接 CAN 总线的 CANL。CANH 和 CANL 间至少要并联一个 120Ω 的终端电阻才能正常通讯。

本驱动器支持多站点通讯，即多台驱动器的 485/CAN 接口通讯线按 A-A、B-B 的方式并联后与一台 485 主站相连。或多台驱动器的 485/CAN 接口通讯线按 CANH-CANH、CANL-CANL 的方式并联后与 CAN 客户端/主站相连。

为了信号更稳定，可将每台驱动器的 COM 连在一起后与 485/CAN 主站的信号地相连。主站可为 PLC、单片机或 PC 机等，485/CAN 主站/CAN 客户端通过每台驱动器设定的不同的地址来对每台驱动器独立操作。

注意：

- 1) 若要使用 485 通讯方式，0x0120 寄存器须配置为 0；此配置值若使用 CAN 通讯，不支持 CANopen 协议；
- 2) 若要使用 CANopen 协议进行 CAN 通讯，0x0120 寄存器须配置为 1，此配置值不支持 485 通讯；
- 3) 在改变 0x0120 寄存器值并保存后，须重新上电，新的通讯方式才会启用；
- 4) 无论 0x0120 配置为 0 或 1，都可以通过将拨码开关 1-8 位全拨到 ON 切换为默认通讯参数通讯方式，此模式下 485 或 CAN 通讯均可用，但不支持 CANopen 协议；
- 5) 在 485/CAN 通讯模式下，若已经通过 485 设备进行通讯，则不能在未掉电情况下通过重新连接到 CAN 设备来进行 CAN 通讯；同样，若已经通过 CAN 设备进行通讯，则不能在未掉电情况下通过重新连接到 485 设备来进行 485 通讯；
- 6) 在 CAN 通讯控制方式下，先断开 CAN 通讯线，再将拨码开关按照由第 8 位到第 1 位的顺序依次拨为 ON 切换为默认通讯参数通讯方式后，再连接上 485 主站是可以进行 485 通讯的；
- 7) 不建议用户在驱动器通电状态下进行第 5)、6) 项这样的接线操作，以免勿操作出现搭线等情况导致驱动器或用户设备损坏。

485 多站点通讯

RS485 多站点通讯示意图如 图 2.7 所示。

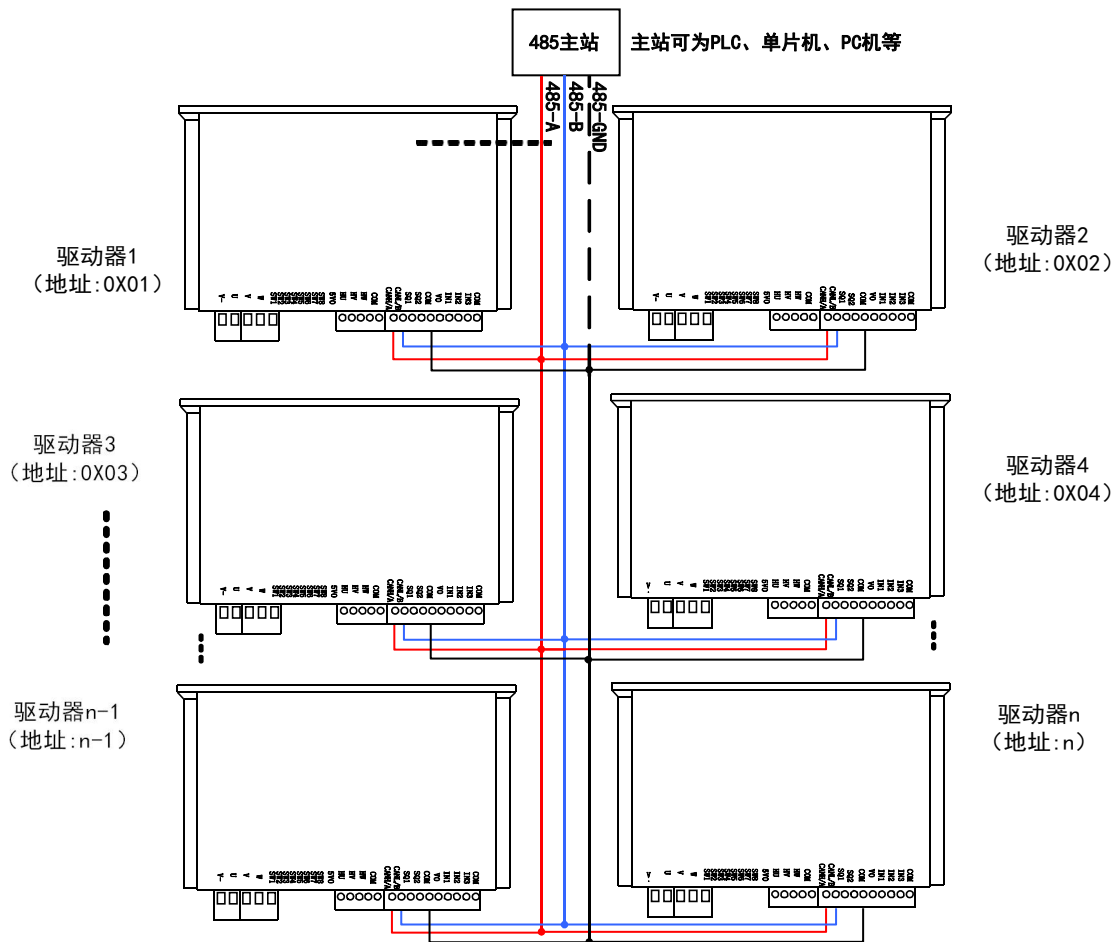


图 2.7 RS485 多站点通讯接线示意图

485/CAN接口通讯线按A-A、B-B分别并联后与 485 主站 485 的信号线A、B连接。并连的每一个驱动器设定的地址应唯一，不能与其它驱动器相同，驱动器地址的配置方法见 表 2.6。485 主站通过通讯帧里的地址字节来指定对哪块驱动器进行操作，配置的地址与通讯帧里指定的地址相同的驱动器才会响应主站的请求(如何配置从站地址见 2.1.5 小节)。如果通讯线较长，可在从站和主站的 485 信号线间各自并联 120Ω 的终端电阻，以消除通讯线中的反射的干扰。

CAN多节点通讯

CAN 多节点通讯示意图如 2.8 所示。

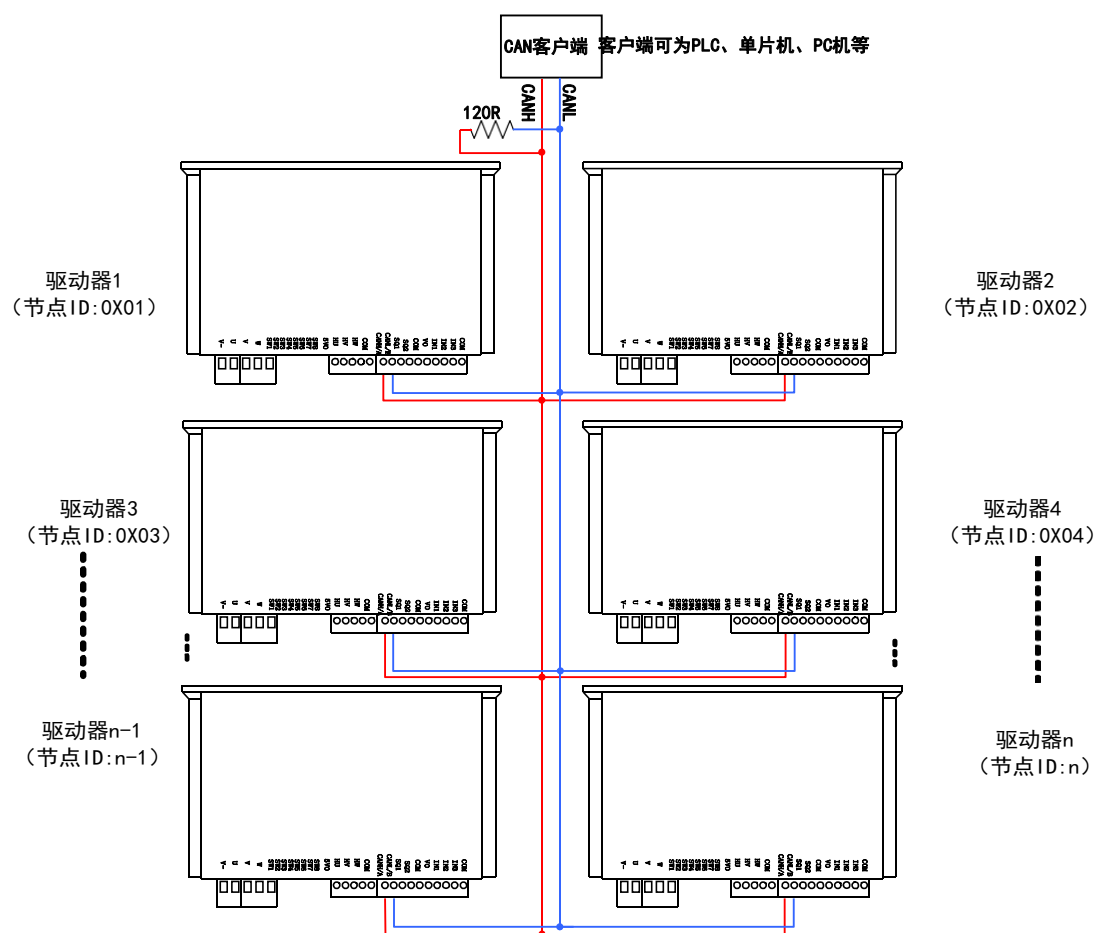


图 2.8CAN 多节点通讯示意图

485/CAN接口通讯线按CANH-CANH、CANL-CANL方式分别并联后与CAN主站/客户端的信号线CANH、CANL分别连接。并连的每一个驱动器设定的节点ID应唯一，不能与其它驱动器相同，驱动器节点ID的配置方法见 表 2.6。CAN主站/客户端通过帧标识位的节点ID位来指定对哪块驱动器进行操作，配置的节点ID与帧标识位里指定的节点ID相同的驱动器才会响应主站/客户端的请求(如何配置节点ID见 2.1.5 小节)。

注意：主站/客户端或从站/服务器至少有一端必须并联一个 120Ω 的电阻才能正常通讯。如果通讯线较长，以消除通讯线中的反射的干扰。但切勿在总线节点较多时每一个节点都并联一个终端电阻，以免导致总线负载过大影响通讯。

状态指示灯

在 485/CAN 通讯模式下，当驱动器绿色指示灯以 0.5Hz 的频率缓慢闪烁时，表示驱动器正处于正常运行状态；当绿色指示灯以 2Hz 的频率较快闪烁时，表示驱动器正处于通讯状态；当红色指示灯单独闪烁时，表示驱动器处于故障状态。

在 CANopen 通讯模式下，当驱动器绿色指示灯常亮时，表示驱动器正处于运行状态；当绿色指示灯以 2Hz 的频率较快闪烁时，表示驱动器正处于预运行状态；当绿色指示灯以 0.5Hz 频率短暂亮一下立刻熄灭时，表示驱动器处于停止状态；当红色和绿色指示灯同时闪烁时表示处于通讯故障状态；当红色指示灯闪烁并且蜂鸣器发出报警声，表示驱动器处于故障状态。

485/CAN通讯控制方式下的使用方法

在使用驱动器前，首先应配置好电机的额定电流参数，如果是初次使用电机，需要先对电机进行学习，再按照相应控制方式要求配置的参数配置好相关参数。具体操作步骤如下：

- 1) 断开驱动器电源。将电机的U、V、W三相电源线接到驱动器电机接口的U、V、W，将电机的霍尔传感器电源线正负极(通常正极为红色，负极为黑色，具体参照电机的相关资料)分别接到驱动器霍尔信号接口的 5VO和COM，霍尔传感器的三霍尔位置信号线接到驱动器霍尔接口的HU、HV、HW，如 图 3.4 所示。

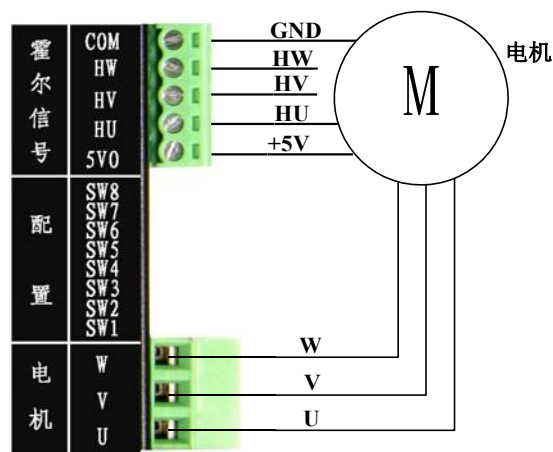


图 3.4 电机接线图

- 2) 将拨码开关第 1~7 位拨到OFF（即拨到上方），第 8 位拨到ON（即拨到下方），这样，驱动器即被配置为通讯控制方式，如 图 3.5 所示，0x009c、0x0120 和 0x0121 寄存器使用默认值 0，从站地址/节点ID被配置为 0x01。

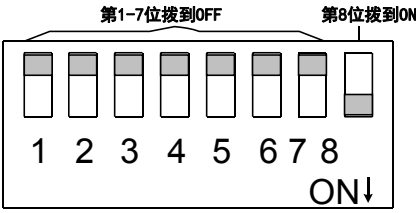


图 3.5 485 通讯控制方式的拨码开关配置

- 3) 将电源的正负极分别接到驱动器电源接口的 V+和 V-;
- 4) 若主站为 485 设备, 485 主站与驱动器的 485/CAN接口按照A-A、B-B的方式连接(为了信号更稳定, 可将驱动器的COM与主站的信号地相连), 如图 3.6 所示, 接通电源(注意: 电源的电压应与电机的额定电压一致, 且能够提供的电流大于电机的额定电流)。
- 5) 通过RS485使用Modbus-RTU 通讯协议与驱动器通讯, 通讯默认波特率为 9600bps, 检验方式为偶校验, 1 位停止位。如果通讯参数被重新配置过, 请使用新配置过的通讯参数进行通讯。

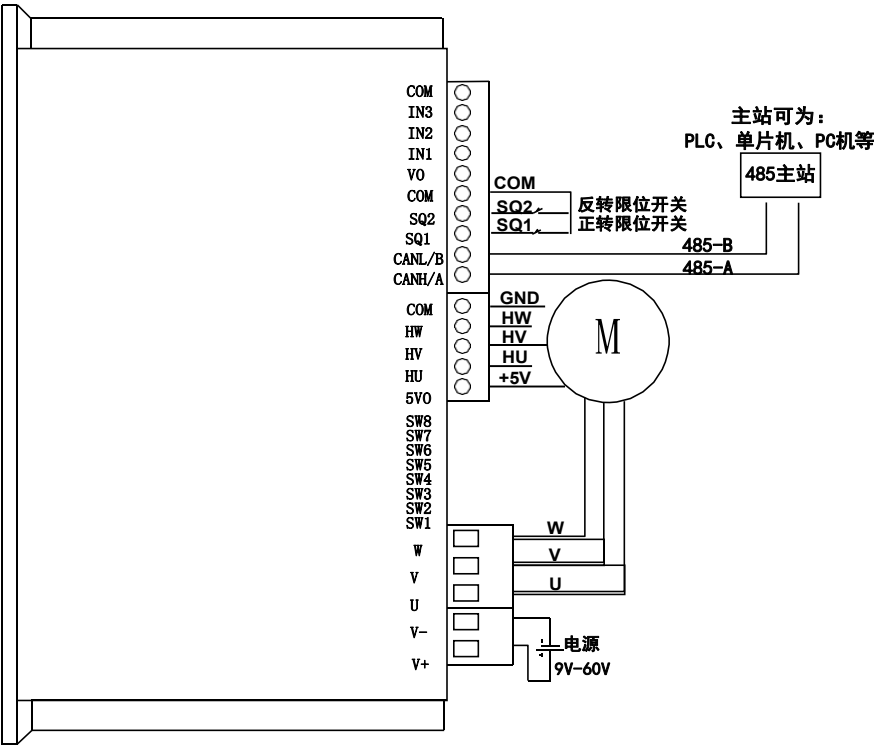


图 3.6 485 通讯电源及电机接线图

- 6) 若主站/客户端为CAN设备, CAN主站/客户端与驱动器的 485/CAN接口按照CANH-CANH、CANL-CANL的方式连接, 并且CAN总线上至少要并联一个 120Ω 的电阻才能正常通讯, 如图 3.7 所示, 接通电源(注意: 电源的电压应与电机的额定电压一致, 且能够提供的电流大于电机的额定电流)。
- 7) CAN通讯默认波特率为 500kbps, 如果通讯参数被重新配置过, 请使用新配置过的通讯参数进行通讯。若要使用CANopen协议, 0x0120 寄存器须配置为 1。通过CAN

可访问CANopen对象字典和Modbus寄存器（如何通过CAN访问Modbus寄存器见7.4节）。

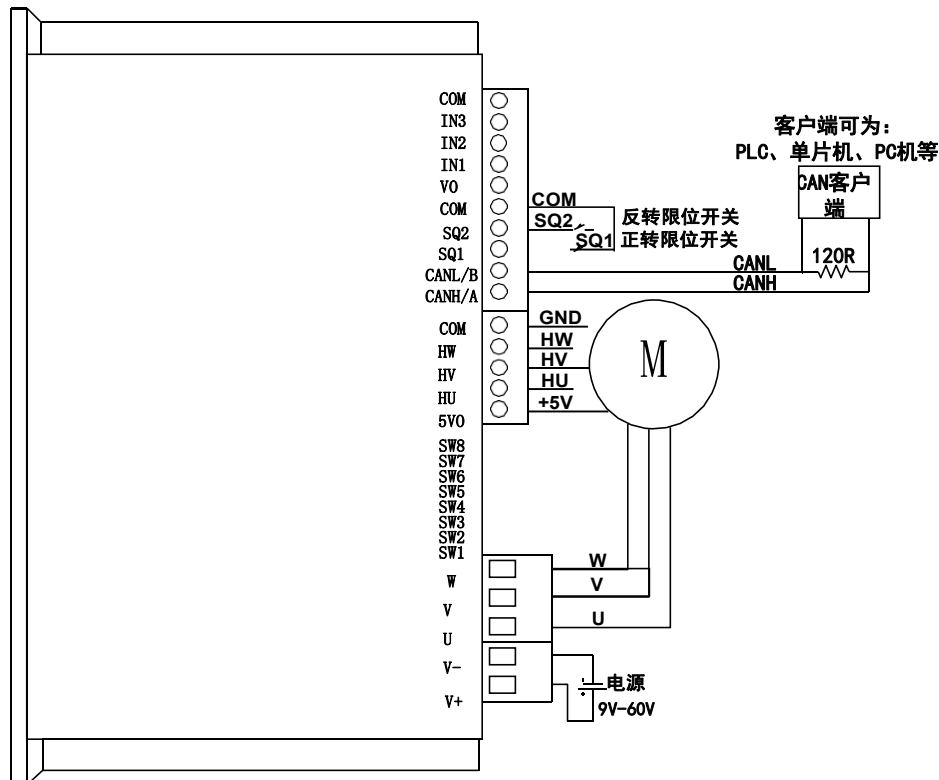


图 3.7 CAN 通讯电源及电机接线图

- 8 通过 0x0050~0x0053 寄存器(详见 6.3.3 小节速度控制寄存器的描述)可临时改变占空比调速方式下PWM的上升、下降缓冲时间及速度闭环和位置闭环方式下的加减速加速度。通过 0x0060~0x0067 寄存器(详见 6.3.4 小节电机控制参数配置寄存器的描述)可配置上电后默认的占空比调速方式下PWM的上升、下降缓冲时间及速度闭环和位置闭环方式下的加减速加速度，以及最大加减速加速度和最大换向频率。
- 9 通过写 0x0042 寄存器设置输出占空比进行占空比调速；通过写 0x0043 寄存器设置电机转动的换向频率(对应转速)进行闭环调速；通过 0x0044 设置位置控制的换向频率(对应转速)，0x0045 寄存器设置位置控制方式为绝对位置还是相对位置，

0x0046 和 0x0047 两个寄存器写入四字节整型的目标位置数值来进行位置闭环控制。可以只操作0x0046 和 0x0047 寄存器或在对0x0046 寄存器写0后只操作0x0047 寄存器来进行位置控制。通过 0x0040 寄存器对电机进行制动操作。0x0040~0x0047 寄存器的描述详见 6.3.3 小节。

- 10) 若使用 CAN 通讯控制方式，还可通过操作 0x2000~0x2003 字典对象对电机进行控制；其中，0x2000 对象设置控制模式为占空比、转速、力矩、位置控制或制动，0x2001 设置控制量，0x2002 设置位置类型，0x2003 设定目标位置。
- 11) 闭环调速的算法可通过 0x0070 寄存器配置为速度闭环控制或时间-位置闭环控制。前者具有超调量小及在高速时调速平稳的特点，但在低速时调速可能不均匀；后者可实现多驱动器对多个电机转动角度的同步控制，以及在低速时调速也平稳，可满足极低速控制的要求，但在调速过程中有一定超调。
- 12) 当闭环调速算法为速度闭环控制时，通过 0x00c0~0x00c5 寄存器配置闭环调速的PID参数；当闭环调速算法为时间-位置闭环控制时，通过 0x00c6~0x00cb寄存器配置闭环调速电机转动时的PID参数，通过 0x00ba~0x00bf寄存器配置闭环调速电机自锁时的PID参数；当为位置闭环控制，也通过 0x00c6~0x00cb 寄存器配置位置闭环控制电机转动时的PID参数，0x00ba~0x00bf配置电机自锁时的PID参数。PID各参数配置过大，可能导致调速或位置控制超调严重甚至出现震荡，PID各参数配置过小可能导致调节缓慢，跟随性差，应合理配置PID参数以使调节效果最佳。PID参数配置相关寄存器详见 6.3.8 小节介绍。
- 13) 通过 0x0080~0x0099 寄存器(详见 6.3.5 小节系统参数配置寄存器的描述)可配置485 通讯控制方式下限位开关触发极性、通讯参数、通讯中断保护时间和堵转停止时间等。

注：也可通过本驱动器配套的 PC 机示例程序进行参数配置及调速控制操作。

各种调速方式的特点

本驱动器可支持占空比调速、力矩控制、速度闭环控制和位置闭环控制(如何配置调速方式见 2.1.4 小节)。各种调速方式的特点如下。

1.1.1 占空比调速

占空比调速方式通过改变等效输出电压来调节电机转速。占空比调速具有响应快的特点，但转速受负载变化有一定变化，在堵转电流不超过配置的最大负载电流的情况下，堵转扭矩与占空比成近似正比，这可表现为当将电机调节为低速转动时，电机扭矩较小。本驱动器另外支持占空比上升/下降缓冲时间配置，以使电机启动/停止过程平稳。

1.1.2 力矩控制

力矩控制方式通过调节输出电流大小来改变电机的扭矩。电机通常工作在堵转状态。力矩控制方式输出的电流可在配置的最大负载电流范围内任意调节。

1.1.3 速度闭环控制

速度闭环控制方式使用 PID 调节算法来对电机进行稳速控制。稳速算法支持速度闭环

控制和时间-位置闭环控制。前者直接对电机转速进行调节，具有超调量小和在高速时调速平稳的特点，但在低速时，可能出现调速不均匀问题；后者通过计算电机随时间改变应该转动的位置来对电机转动位置进行控制，从而间接对电机进行了稳速控制，此方式可满足多台驱动器对多个电机转动位置进行同步控制的要求以及超低速稳速控制的要求，但转速调节有一定超调。本驱动器支持闭环调速加速度配置，对于使用速度闭环控制算法，可将加速配置大一些，以使稳速响应更快；而对于使用时间-位置闭环控制算法，加速度配置过大则可能导致超调严重或切换电机转动方向过程不平稳。

1.1.4 位置闭环控制

位置闭环控制使用 PID 调节算法来对电机转动位置进行控制。当给定目标位置后，驱动器会根据配置的加速加速度、减速加速度和最大速度，自动计算电机运行过程中当前转动位置的目标实时速度并进行调控，从而使电机按照配置的速度和加速度参数准确地转动到目标位置。在对电机位置进行调控过程中，驱动器也能同时估算出电机转动到目标位置所需要的时间。注意，如果加速度配置过大或制动电流配置过小可能导致驱动器提供不了所需的加速度而使位置控制出现超调，因此应合理配置加速度。

在数字/模拟信号控制方式下，驱动器可实现固定行程内的电机转动位置调节以及使用脉冲信号对电机进行步进控制；在 485 通讯控制方式下，可实现对电机绝对转动位置和相对转动位置的控制。

通讯控制方式的接法和配置

485 通讯控制

此用法通过 485 通讯实现对电机的控制操作。485 通讯控制的接法如 图 4.57 所示。485 主站(主站可以是PLC、单片机或PC机等)的 485 两信号线按照A-A、B-B的方式与驱动器的 485 接口相连。485 主站通过Modbus-RTU通讯协议操作驱动器的相关寄存器对电机进行调速、方向控制、位置控制等操作。在 485 通讯控制方式下，驱动器支持占空比调速、速度闭环控制和位置闭环控制。

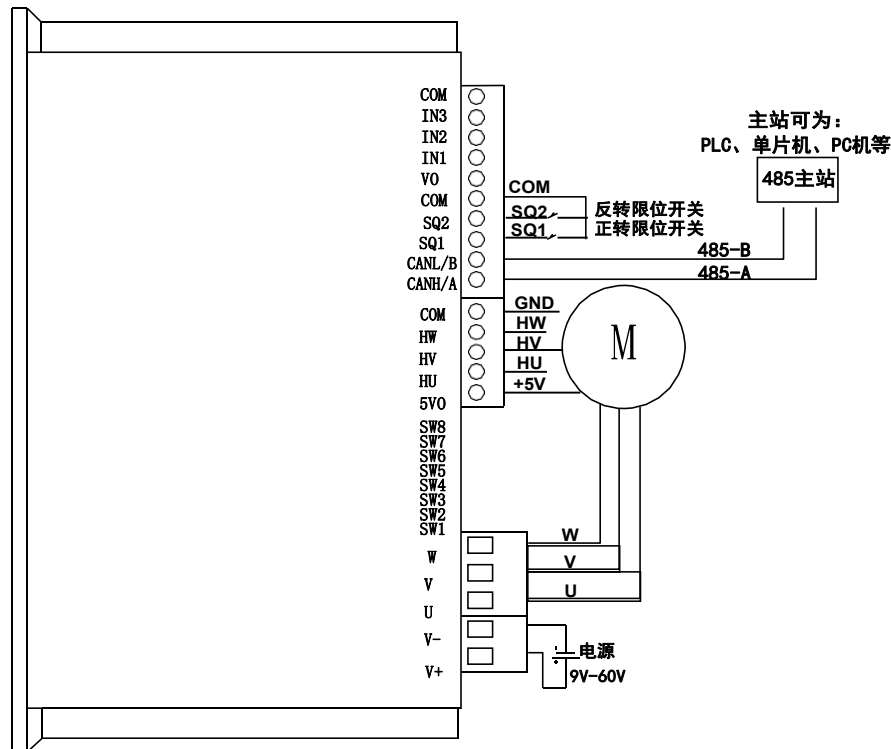


图 4.57 485 通讯控制的接法

使用RS485 与驱动器通讯时，通讯参数及设备地址应与驱动器一致。通讯参数包括波特率、奇偶效验方式和停止位。驱动器默认的通讯参数为，波特率 9600bps，偶校验，1 停止位。驱动器的波特率可通过 0x0090 和 0x0091 寄存器进行配置，驱动器支持的波特率范围为 1200~115200bps；校验方式和停止位通过 0x0092 寄存器进行配置，驱动器支持偶校验+1 停止位、奇校验+1 停止位和无校验+2 停止位。通讯参数相关寄存器详见 6.3.5 小节系统参数配置寄存器。驱动器的Modbus从站设备地址通过拨码开关第 1~7 位配置（从站地址译码表见 表 2.6）或由 0x009c寄存器指定；第 8 位为控制方式位，使用 485/CAN通讯控制时第 8 位应拨到ON；拨码开关的配置如 图 4.58 所示。

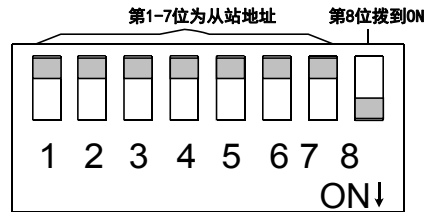


图 4.58 485 通讯方式拨码开关的配置

占空比调速方式下PWM的上升、下降缓冲时间及速度闭环和位置闭环方式下的加减速加速度可通过 0x0050~0x0053 寄存器(详见 6.3.3 小节速度控制寄存器的描述)临时单独改变;而上电时默认的占空比调速方式下PWM的上升、下降缓冲时间及速度闭环和位置闭环方式下的加减速加速度,以及最大加减速加速度和最大换向频率通过 0x0060~0x0067 寄存器(详见 6.3.4 小节电机控制参数配置寄存器的描述)进行配置。

通过写 0x0042 寄存器设定输出占空比进行占空比调速;通过写 0x0043 寄存器设定电机转动的换向频率(对应转速)进行闭环调速;通过 0x0044 设定位置控制的换向频率(对应转速), 0x0045 寄存器设定位置控制方式为绝对位置还是相对位置, 0x0046 和 0x0047 两个寄存器写入四字节整型的目标位置数值来进行位置闭环控制;通过 0x0040 寄存器对电机进行制动操作。占空比调速、闭环调速、位置控制三种调速方式间可直接切换,写各调速方式对应的输出量寄存器(如 0x0042、0x0043、0x0047 寄存器等)即可切换为相应的调速方式。对于位置控制调速方式,可以只操作 0x0046 和 0x0047 寄存器或在对 0x0046 寄存器单次写 0 后只操作 0x0047 寄存器来进行位置控制。0x0040~0x0047 寄存器的描述详见 6.3.3 小节。

闭环调速的算法可通过 0x0070 寄存器配置为速度闭环控制或时间-位置闭环控制。前者具有超调量小及在高速时调速平稳的特点,但在低速时调速可能不均匀;后者可实现多驱动器对多个电机转动角度的同步控制,以及在低速时调速也平稳,可满足极低速控制的要求,但在调速过程中有一定超调。

当闭环调速算法为速度闭环控制时,通过 0x00c0~0x00c5 寄存器配置闭环调速的PID参数;当闭环调速算法为时间-位置闭环控制时,通过 0x00c6~0x00cb 寄存器配置闭环调速电机转动时的PID参数,通过 0x00ba~0x00bf 寄存器配置闭环调速电机自锁时的PID参数;当为位置闭环控制,也通过 0x00c6~0x00cb 寄存器配置位置闭环控制电机转动时的PID参数, 0x00ba~0x00bf 配置电机自锁时的PID参数。PID各参数配置过大,可能导致调速或位置控制超调严重甚至出现震荡, PID各参数配置过小可能导致调节缓慢,跟随性差,应合理配置PID参数以使调节效果最佳。PID参数配置相关寄存器详见的 6.3.8 介绍。

通过 0x0080~0x0099 寄存器(详见 6.3.5 小节系统参数配置寄存器的描述)可配置 485 通讯控制方式下限位开关触发极性、通讯参数、通讯中断保护时间和堵转停止时间等。我们通过 0x0095 寄存器设置通讯中断保护时间,当在设置的时间内没有对驱动器进行通讯访问时,驱动器便会进行制动操作,这样可解决机械装置运动过程中,通讯线路出现故障导致机械装

置不受主站控制问题，我们可以将实时状态寄存器(详见 6.3.2 小节)作为周期性查询访问的寄存器。我们通过 0x008e寄存器设置堵转停止时间，当电机堵转时电流达到配置的最大负载电流且电机转速为 0，当这种状态持续时间达到配置的堵转停止时间后，驱动器将进行制动，堵转停止的状态可通过 0x0032 寄存器读取，我们可以通过制动或反转操作清除堵转停止标志。

通过 0x0020~0x0034 寄存器(详见 6.3.2 小节实时状态寄存器的描述)我们可读取输出 PWM值、电机换向频率、电机相电流、电机转动位置、电机转速等电机相关实时状态值以及各输入信号的实时数值。通过 0x0020 寄存器读取PWM输出值，PWM输出值可反映驱动器输出加在电机相线上的电压，相电压约等于电源电压乘以占空比。通过 0x0022 寄存器读取电机换向频率，电机换向频率为电机转动时霍尔传感器输出的霍尔信号改变的频率，单位为 Hz。电机相电流为电机U、V、W三相线中电流的平均值，单位为A。通过 0x0024 和 0x0025 寄存器读取电机转动位置为电机朝着某一方向转动的换向次数(或霍尔脉冲数)，电机位置控制的预计完成时间可通过 0x0026 和 0x0027 寄存器读取，完成状态通过 0x0023 寄存器读取。电机转速通过 0x0034 寄存器读取，电机转速为测量的电机实时转速，单位为RPM，要使读取的电机实时转速与电机真实转速一致，那么应先通过 0x0073 和 0x0074 寄存器配置电机极个数和减速比。

485 通讯控制方式主要相关寄存器如 表 4.57 所示。

表 4.57 485 通讯控制方式相关寄存器的配置

寄存器地址	寄存器作用	值	描述
0x0040	电机制动控制	0, 1, 2	0: 正常停止 1: 紧急制动 2: 自由停止
0x0042	设定占空比	-1000~1000	数值乘以 0.1%为目标占空比
0x0043	设定速度闭环控制目标速度	-32768~32767	数值乘以 0.1 为目标换向频率，单位为 Hz
0x0044	设定位置闭环控制行走速度	-32768~32767	数值乘以 0.1 为目标换向频率，单位为 Hz
0x0045	设定位置闭环控制类型	0,1	0: 绝对位置 1: 相对位置
0x0046-0x0047	设定位置闭环控制目标位置	-2147483648 ~ 2147483647	
0x0050	临时设定占空比调速加速缓冲时间	0~255	数值乘以 0.1 为输出占比空由 0 增加到 100.0%所需时间
0x0051	临时设定占空比调速减速缓冲时间	0~255	数值乘以 0.1 为输出占比空由 100.0%减小到 0 所需时间
0x0052	临时设定速度闭环控制、位置闭环控制加速加速度	0~66635	数值乘以 0.1 为换向频率增大速度，单位为 Hz/s
0x0053	临时设定速度闭环控制、位置闭环控制减速加速度	0~66635	数值乘以 0.1 为换向频率减小速度，单位为 Hz/s

	度		
0x006a	配置电机额定电流	0~3000	数值乘以 0.01 为电流值，单位为A。
0x006b	配置电机最大负载电流	0~3000	数值乘以 0.01 为电流值，单位为A。
0x006c	配置电机最大制动电流	0~600	数值乘以 0.01 为电流值，单位为A。
0x0070	配置速度闭环控制算法	0, 1	0: 速度闭环控制 1: 时间-位置闭环控制
0x0071	配置位置闭环控制允许误差	0~65535	
0x0072	配置位置闭环控制超调后是否修正	0, 1	0: 不进行修正 1: 进行修正
0x0073	配置电机极个数	0~65535	设定电机极个数，电机极个数通常为 3 的倍数
0x0074	配置电机减速比	0~65535	
0x0020	PWM 输出值	0~1000	数值乘以 0.1%为占空比
0x0021	实时电机相电流	0~3000	数值乘以 0.01 为电流值，单位为A。
0x0022	实时电机换向频率	-32768~32767	当 0x0035 寄存器为 1 时，数值即为换向频率；当 0x0035 寄存器为 0 时，数值乘以 0.1 为换向频率；单位为 Hz；换向频率除以电机极个数再乘以 20 为电机转速，单位 RPM。
0x0023	位置控制完成状态	0, 1	0: 未完成 1: 完成
0x0024-0x0025	电机实时位置	-2147483648 ~ 2147483647	
0x0026-0x0027	位置控制预计剩余完成时间	0~4294967295	单位为ms
0x0032	电机堵转状态	0, 1, 2	0: 未堵转 1: 正转堵转停止 2: 反转堵转停止
0x0033	错误状态	0, 1, 2, 3, 4	0: 无错误 1: 尚未学习 2: 堵转停止 3: 霍尔错误 4: 无法达到目标速度
0x0034	电机实时转速	0~65535	当 0x0035 寄存器为 1 时，数值乘以 10 为转速；当 0x0035 寄存器为 0 时，数值即为转速；单位为 RPM。 注：需先通过 0x0073 和 0x0074 寄存器配

			置正确的电机极个数和减速比，读取的转速才正确。
0x0035	转速是否需要乘以 10	0, 1	0: 数值即转速; 1: 数值乘以 10 为转速

CAN通讯控制方式

此用法通过CAN通讯实现对电机的控制操作。CAN通讯控制的接法如图 4.57 所示。CAN主站(客户端)(主站可以是PLC、单片机或PC机等)的通讯线按照CANH-CANH、CANL-CANL的方式与驱动器的 485/CAN通讯接口相连。CAN主站(客户端)通过访问驱动器的CANOpen对象字典或Modbus寄存器对电机进行调速、方向控制、位置控制等操作。在CAN通讯控制方式下，驱动器支持占空比调速、力矩控制、速度闭环控制和位置闭环控制。

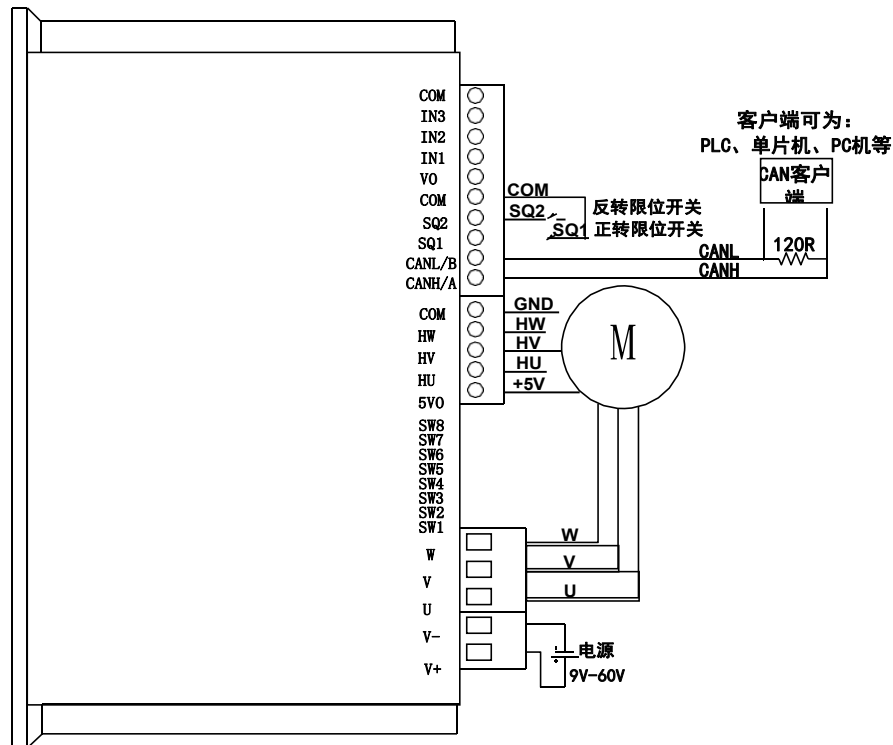


图 4.59 CAN 通讯控制的接法

通过CAN既可访问CANOpen对象字典也可访问Modbus寄存器。若通过CANOpen的SDO访问Modbus寄存器，若Modbus寄存器地址在 0x0000~0x0fff范围，则访问相应寄存器的CANOpen索引号为：0x4000+寄存器地址；若Modbus寄存器地址在 0x7000~0x7fff范围，则访问相应寄存器的CANOpen索引号为：寄存器地址-0x2000；子索引号固定为 0；通过 0x2B 和 0x23 命令分别对双字节和四字节数据的寄存器进行写操作；通过 0x40 命令对寄存器进行读操作。CAN通讯协议详见第 7 章节。

使用CAN与驱动器通讯时，波特率及节点ID应与驱动器设置一致。驱动器默认的CAN波特率为 500kbps。可通过 0x0122 寄存器配置CAN波特率，驱动器支持的波特率范围为 10kbps~1Mbps。通讯参数相关寄存器详见 6.3.11 小节CAN配置参数寄存器。驱动器的CAN节点ID可通过拨码开关第 1~7 位配置（节点ID译码表见 表 2.6）或由 0x0121 寄存器指定；第 8 位为控制方式位，使用485/CAN通讯控制时第8 位应拨到ON；拨码开关的配置如图 4.58 所示。

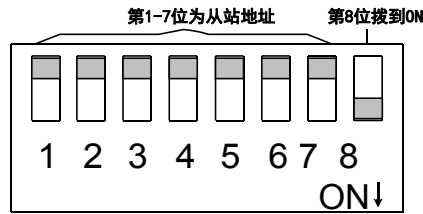


图 4.60 485 通讯方式拨码开关的配置

在使用电机前，应首先对电机的额定电流和工作电流进行配置。可通过 0x006a和 0x006b 寄存器(详见 6.3.4 小节)配置电机的额定电流和最大负载电流，配置的电机额定电流应与电机实际额定电流一致或略高，最大负载电流可用来配置电机的最大负载/堵转力矩，如无要求，通常与额定电流配置相同，制动电流通过与电机额定电流配置一致。电机额定电流可从电机的铭牌标示或数据手册上获得。

占空比调速方式下PWM的上升、下降缓冲时间及速度闭环、位置闭环方式下的加减速加速度以及速度闭环或位置闭环控制方式的最大速度可通过访问对象字典 0x2004~0x2008 索引（详见 7.6.1 小节电机控制对象的描述）临时单独改变；而上电时默认的占空比调速方式下PWM的上升、下降缓冲时间及速度闭环和位置闭环方式下的加减速加速度，以及最大加减速加速度和最大换向频率通过 0x0060~0x0067 寄存器(详见 6.3.4 小节电机控制参数配置寄存器的描述)进行配置。

通过对象字典 0x2000 索引可设定电机控制类型为占空比调速、闭环调速、力矩控制、位置控制以及电机制动；通过对象字典 0x2001 索引可设定占空比调速、闭环调速和位置控制的速度控制量以及力矩控制方式的电流控制量；通过对象字典 0x2002 和 0x2003 索引可设定位置控制方式的位置类型（绝对位置或相对位置）以及目标位置值。

通过对象字典 0x200a 索引可设置速度单位为 Hz 或 RPM，通过 0x200b 索引可设置加速度单位为 Hz/s 或 Rad/s²。注意：须先通过 0x0073 寄存器配置电机极个数后才可设置单位为 RPM 和 Rad/s²。

闭环调速的算法可通过 0x0070 寄存器配置为速度闭环控制或时间-位置闭环控制。前者具有超调量小及在高速时调速平稳的特点，但在低速时调速可能不均匀；后者可实现多驱动器对多个电机转动角度的同步控制，以及在低速时调速也平稳，可满足极低速控制的要求，但在调速过程中有一定超调。

当闭环调速算法为速度闭环控制时，通过 0x00c0~0x00c5 寄存器配置闭环调速的PID参数；当闭环调速算法为时间-位置闭环控制时，通过 0x00c6~0x00cb寄存器配置闭环调速电机转动时的PID参数，通过 0x00ba~0x00bf寄存器配置闭环调速电机自锁时的PID参数；当为位置闭环控制，也通过 0x00c6~0x00cb 寄存器配置位置闭环控制电机转动时的PID参数，0x00ba~0x00bf配置电机自锁时的PID参数。PID各参数配置过大，可能导致调速或位置控制超调严重甚至出现震荡，PID各参数配置过小可能导致调节缓慢，跟随性差，应合理配置PID参数以使调节效果最佳。PID参数配置相关寄存器详见的 6.3.8 介绍。

通过 0x0080~0x0099 寄存器(详见 6.3.5 小节系统参数配置寄存器的描述)可配置 485 通讯控制方式下限位开关触发极性、通讯参数、通讯中断保护时间和堵转停止时间等。我们通过 0x0095 寄存器设置通讯中断保护时间，当在设置的时间内没有对驱动器进行通讯访问时，驱动器便会进行制动操作，这样可解决机械装置运动过程中，通讯线路出现故障导致机械装

置不受主站控制问题。我们通过 0x008e寄存器设置堵转停止时间，当电机堵转时电流达到配置的最大负载电流且电机转速为 0，当这种状态持续时间达到配置的堵转停止时间后，驱动器将进行制动，堵转停止的状态可通过对象字典 0x2111 索引读取，我们可以通过制动或反转操作清除堵转停止标志。

通过对象字典的 0x2100~0x2122 索引(详见 7.6.2 小节实时状态对象字典的描述)我们可读取输出PWM值、电机换向频率、电机相电流、电机转动位置、电机转速等电机相关实时状态值以及各输入信号的实时数值。通过 0x2101 索引读取PWM输出值，PWM输出值可反映驱动器输出加在电机相线上的电压，相电压约等于电源电压乘以占空比。通过 0x2102 索引读取电机换向频率，电机换向频率为电机转动时霍尔传感器输出的霍尔信号改变的频率，单位为Hz。通过 0x2100 索引读取的数值乘以 0.01 为电机相电流，电机相电流为电机U、V、W三相线中电流的平均值，单位为A。通过 0x2105 索引读取电机转动位置为电机朝着某一方向转动的换向次数(或霍尔脉冲数)，电机位置控制的预计完成时间可通过 0x210b索引读取，完成状态通过 0x2106 索引读取。电机转速通过 0x210a索引读取，电机转速为测量的电机实时转速，单位为RPM，要使读取的电机实时转速与电机真实转速一致，那么应先通过 0x0073 和 0x0074 寄存器配置电机极个数和减速比。

485 通讯控制方式主要相关寄存器如 表 4.57 所示。

表 4.58 CAN 通讯控制方式相关 Modbus 寄存器配置

寄存器地址	寄存器作用	值	描述
0x006a	配置电机额定电流	0~3000	数值乘以 0.01 为电流值，单位为 A。
0x006b	配置电机最大负载电流	0~3000	数值乘以 0.01 为电流值，单位为 A。
0x006c	配置电机最大制动电流	0~600	数值乘以 0.01 为电流值，单位为 A。
0x0070	配置速度闭环控制算法	0, 1	0: 速度闭环控制 1: 时间-位置闭环控制
0x0071	配置位置闭环控制允许误差	0~65535	
0x0072	配置位置闭环控制超调后是否修正	0, 1	0: 不进行修正 1: 进行修正
0x0073	配置电机极个数	0~65535	设定电机极个数，电机极个数通常为 3 的倍数
0x0074	配置电机减速比	0~65535	

表 4.59 CAN 通讯控制方式电机控制相关对象字典

索引号	子索引	对象作用	类型	值	权限	描述
0x2000	0	电机控制类型	U8	0x00~0x03, 0x10~0x12	RW/E	0x00: 占空比调速 0x01: 闭环调速 0x02: 力矩控制 3: 位置闭环 0x10: 正常停止 0x11: 紧急停止 0x12: 自由停止
0x2001	0	电机控制量	S16 /S32	占空比: -1000~1000	RW	占空比调速方式时，写入数值乘以 0.1%为输出占空比；

				速度/位置闭环： -最大速度~最大速度 力矩控制： -最大电流~最大电流		闭环调速或位置控制时，当 0x200A 索引对象数值为 1 时，写入数值为电机目标转速(RPM)；当 0x200A 索引对象数值为 0 时，写入数值为电机目标换向频率(Hz)，若 0x0077 寄存器值为 0，则设写入数值乘以 0.1 为目标换向频率； 力矩控制方式时，写入数值乘以 0.01 为目标电流 (A)。
0x2002	0	位置类型	U8	0,1	RW	0: 绝对位置 1: 相对位置
0x2003	0	目标位置	S32		RW	
0x2004	0	开环调速 PWM 上升缓冲时间	U8	0~255	RW	数值乘以 0.1 为输出占比空由 0 增加到 100.0%所需时间
0x2005	0	开环调速 PWM 下降缓冲时间	U8	0~255	RW	数值乘以 0.1 为输出占比空由 100.0%减小到 0 所需时间
0x2006	0	闭环调速加速加速度	U16	1~65535	RW	数值乘以 0.1 为换向频率增大速度，单位为 Hz/s
0x2007	0	闭环调速减速加速度	U16	1~65535	RW	数值乘以 0.1 为换向频率减小速度，单位为 Hz/s
0x2008	0	最大速度	U16	1~65535	RW	当 0x200A 索引对象数值为 1 时，写入数值为电机最大转速(RPM)；当 0x200A 索引对象数值为 0 时，写入数值为电机最大换向频率 (Hz)，若 0x0077 寄存器值为 0，则设写入数值乘以 0.1 为换向频率；
0x200A	0	速度单位	U8	0,1	RW/E	0: Hz 1: RPM 若要设置速度单位为 RPM，须先通过 0x0073 寄存器配置电机极个数
0x200B	0	加速度单位	U8	0,1	RW/E	0: Hz/s 1: Rad/s ² 若要设置加速度单位为 Rad/s ² ，须先通过 0x0073 寄存器配置电机极个数
0x200F	0	重新设定电机转动位置计数值	S32		WO	若写0 则对当前计数值清零
0x2100	0	电机相电流	U16		RO	数值乘以 0.01 为电流值，单

						位为A。
0x2101	0	输出 PWM	S16	-1000~1000	RO	数值乘以 0.1%为占空比
0x2102	0	电机换向频率	S32		RO	数值乘以 0.1 为换向频率
0x2105	0	电机转动位置计 数值	S32		RO	
0x2106	0	电机位置控制完 成状态	U8	0,1	RO	0: 未完成 1: 完成
0x210a	0	电机转速	U32		RO	单位 RPM。
0x210b	0	完成位置控制剩 余时间	U32		RO	单位为ms
0x2111	0	电机堵转状态	U8	0,1,2	RO	0: 未堵转 1: 正转堵转停止 2: 反转堵转停止
0x2112	0	故障状态	U8	0~10	RO	0: 无错误 1: 尚未学习 2: 堵转停止 3: 霍尔错误 4: 无法达到目标速度 5: 保留 6: 过流关断 7: 过热关断 8: 过压关断 9: 欠压关断 10: 短路/过流关断

注：/E 表示可通过 0x1010 索引保存到内部存储器中。

485/CAN多站点通讯控制

此用法使用一个 485/CAN主站(CAN客户端) (主站可以是PLC、单片机或PC机等)通过 485/CAN通讯方式操作多台驱动器，从而实现对多个电机的分别控制。拓扑图如 图 4.61 所示。485/CAN多站点通讯的接法见 5.4 小节。

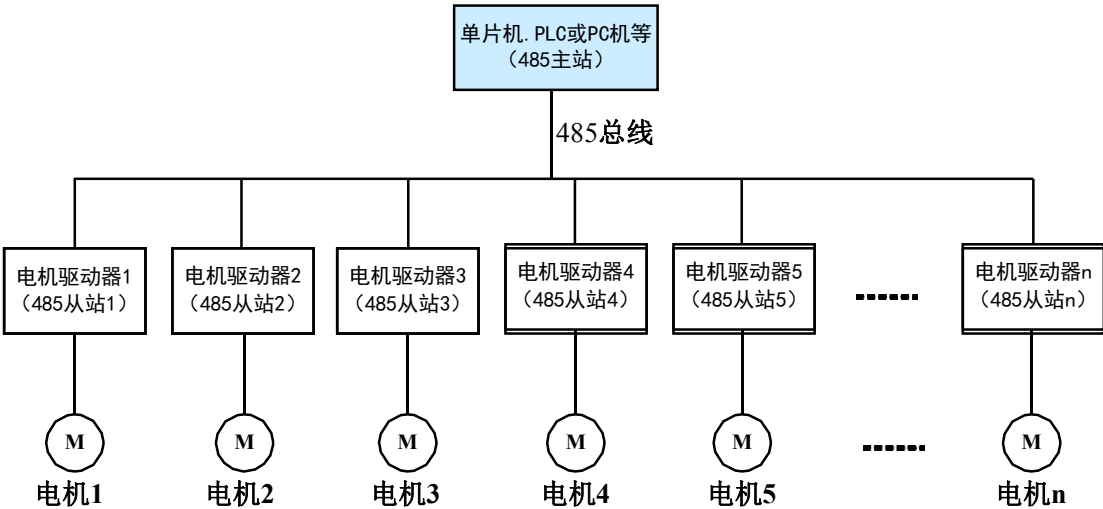


图 4.61 485 多站点通讯控制拓扑图

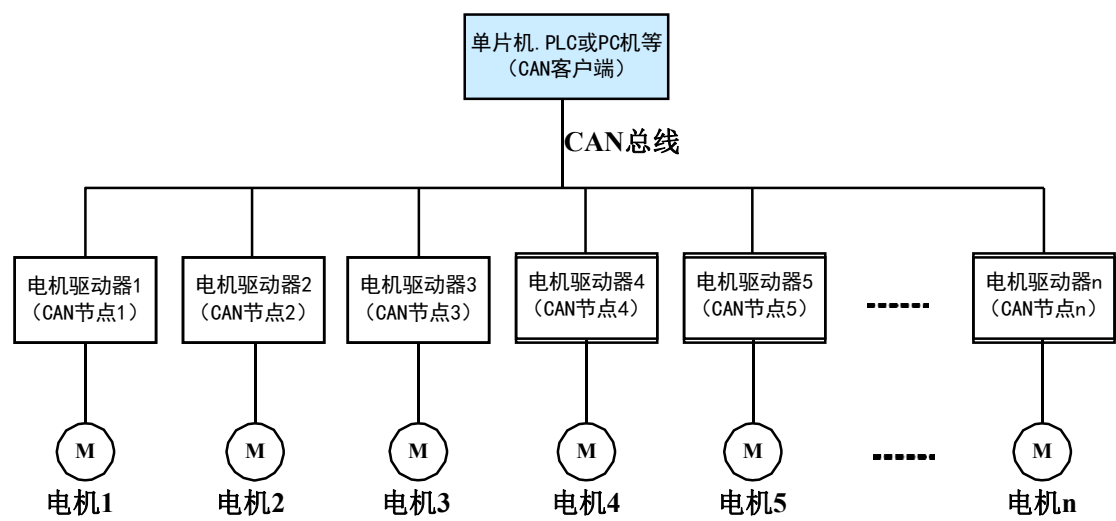


图 4.62 CAN 多站点通讯控制拓扑图

485/CAN多站点通讯控制

若主站为 485 设备，则所有驱动器的 485/CAN的信号线A、B分别并联后与 485 主站 485 的信号线A、B连接。485/CAN多站点通讯示意图如 图 5.13 所示。

为了信号更稳定，可将每台驱动器的COM相连后与主站的信号地相连。每一个驱动器配置的从站地址应唯一，不能与其它驱动器相同，驱动器从站地址的配置方法见 表 2.6。485 主站通过通讯帧里的地址字节来指定对哪块驱动器进行操作，配置的地址与通讯帧里指定的地址相同的驱动器才会响应主站的请求(如何配置从站地址见 2.1.5 小节)。如果通讯线较长，可在从站和主站的 485 信号线间各自并联 120Ω 的终端电阻，以消除通讯线中的反射的干扰。

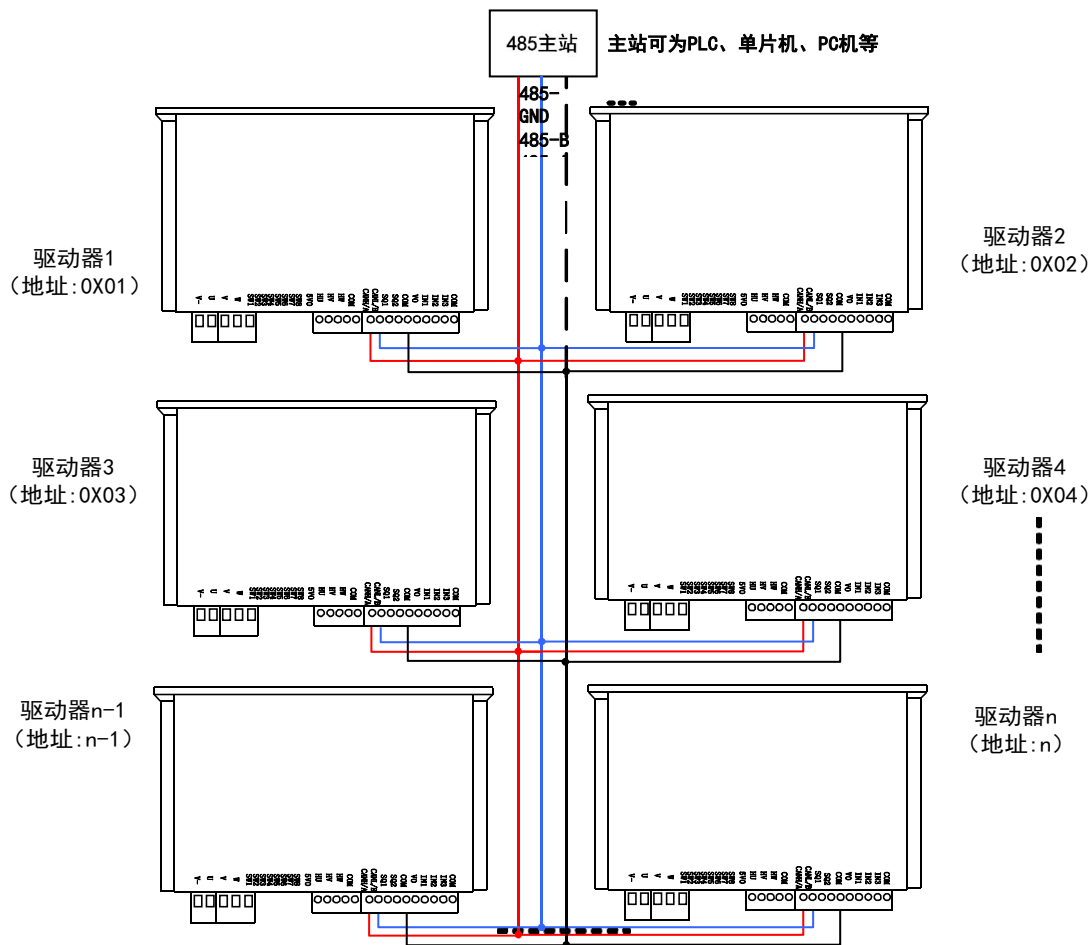


图 5.13 RS485 多站点通讯接线示意图

若主站/客户端为CAN设备，则所有驱动器的 485/CAN的信号线CANH、CANL分别并联后与CAN主站/客户端的CAN的信号线CANH、CANL连接。在CAN总线上，至少并联一个 120Ω 的终端电阻才能正常通讯。如 图 5.14 所示。

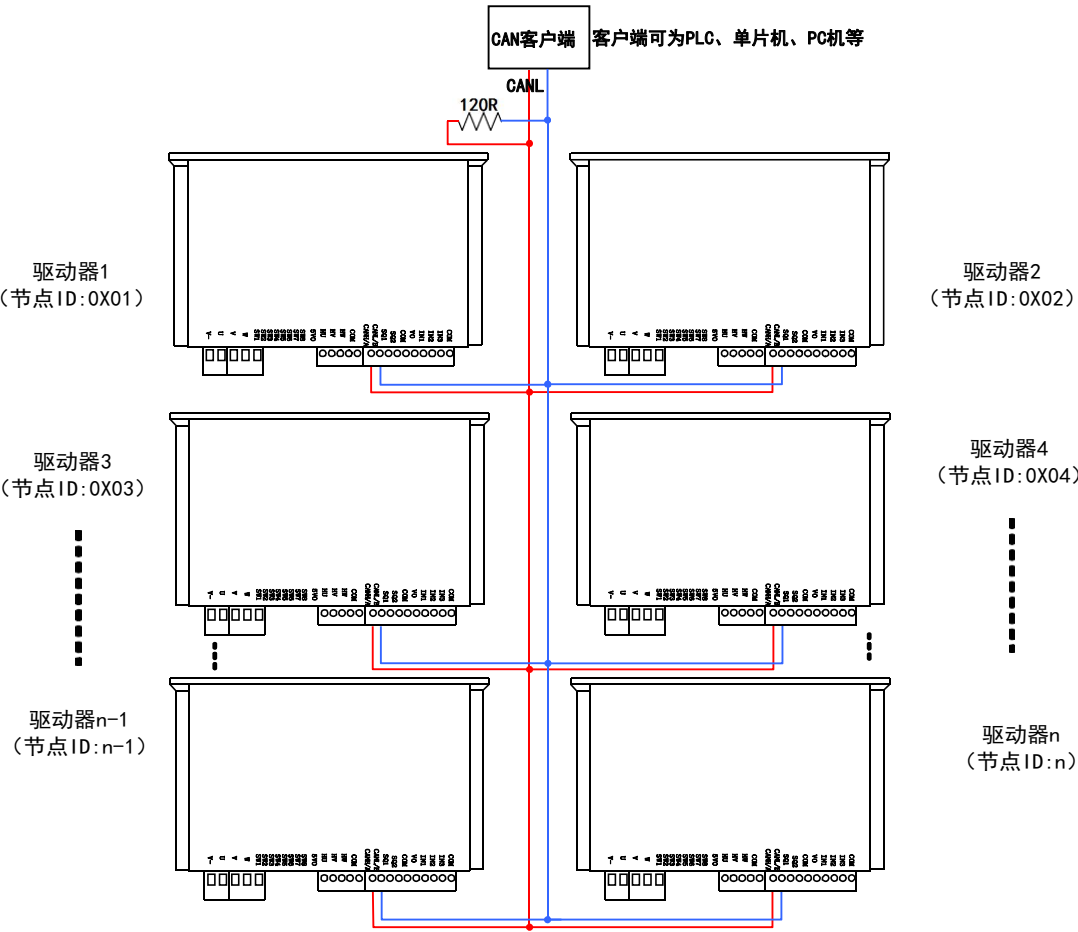


图 5.14 CAN 多节点通讯控制

每一个驱动器配置的节点ID应唯一，不能与其它驱动器相同，驱动器节点ID的配置方法见 2.1.5 小节。CAN主站/客户端通过帧标识的节点ID位来指定对哪块驱动器进行操作，配置的节点ID与帧标识指定的节点ID相同的驱动器才会响应主站的请求。

如果通讯线较长，可在总线适当位置增加并联 120Ω 的终端电阻，以消除通讯线中的反射的干扰。

RS485 通讯协议

本驱动器 485 通讯方式使用MODBUS-RTU(国标GB/T19582-2008)通讯协议，支持一主站控制多个从站，通过拨码开关或Modbus寄存器可配置 127 个从站地址，主站可以是单片机、PLC或PC机等。关于从站地址的配置见 2.1.5 小节。

RS485 通讯参数

当 0x0120 寄存器值为 0（默认值）或拨码开关第 1~8 位全为 ON（默认通讯参数通讯）时，驱动器通讯方式为 485/CAN，用户可通过 485 设备与驱动器进行通讯。

数字/模拟信号控制方式时（拨码开关第 8 位为 OFF），若 0x009c 寄存器值为 0（默认值），则串口波特率固定为 9600bps，数据位为 8 位，偶校验，停止位为 1 位；从站地址固定为 0x01；若通过 0x009c 寄存器指定从站地址且 0x009d 寄存器值配置为 1，则通讯参数同 485 通讯控制方式。

485 通讯控制方式时（拨码开关第 8 位为 ON），波特率默认为 9600bps，数据位为 8 位，偶校验，停止位为 1 位；波特率可配置范围 1200-115200bps，数据位固定为 8 位，校验方式可配置为奇校验、偶校验或无校验，当为奇、偶校验时停止位为 1 位，无校验时停止位为 2 位；从站地址由拨码开关 1~7 位设定或通过 0x009c 寄存器指定。

每个字符使用 11 个比特位(1 位起始位，8 位数据位，1 位校验位加 1 位停止位或无校验位加 2 位停止位)；当波特率在 19200bps 及以下时，字符超时时间为 1.5 个字符间隔；19200bps 以上时，超时时间为 0.75ms；当发生字符超时，之前接收到的数据将被视为无效；帧超时时间为 3.5 个字符间隔，当发生帧超时，就表示这一帧发送完成。

MODBUS-RTU帧格式

本驱动器支持 MODBUS 的 0x03(读保持寄存器)、0x06(写单个寄存器)、0x10(写多个寄存器)和 0x2B(读设备识别码)功能码。

0x03 读保持寄存器

主站发送：

字节	1	2	3	4	5	6	7	8
内容	ADR	0x03	起始寄存器高字节	起始寄存器低字节	寄存器数高字节	寄存器数低字节	CRC 低字节	CRC 高字节

第 1 字节 ADR：从站地址码（=001~254）

第 2 字节 0x03：读寄存器值功能码

第 3、4 字节：要读的寄存器开始地址

第 5、6 字节：要读的寄存器数量

第 7、8 字节：从字节 1 到 6 的 CRC16 校验码

从站回送：

字节	1	2	3	4、5	6、7		M-1、M	M+1	M+2
内容	ADR	0x03	字节总数	寄存器数据 1	寄存器数据 2	...	寄存器数据 M	CRC 低字节	CRC 高字节

第 1 字节 ADR: 从站地址码 (=001~254)
 第 2 字节 0x03: 返回读功能码
 第 3 字节: 从 4 到 M (包括 4 及 M) 的字节总数
 第 4 到 M 字节: 寄存器数据
 第 M+1、M+2 字节: 从字节 1 到 M 的 CRC16 校验码
 当从站接收错误时, 从站回送:

字节	1	2	3	4	5
内容	ADR	0x83	异常码	CRC 低字节	CRC 高字节

第 1 字节 ADR: 从站地址码 (=001~254)
 第 2 字节 0x83: 读寄存器值出错
 第 3 字节 异常码: 见 6.2.4 小节
 第 4、5 字节: 从字节 1 到 3 的 CRC16 校验码

0x06 写单个寄存器

主站发送:

字节	1	2	3	4	5	6	7	8
内容	ADR	0x06	寄存器高字节地址	寄存器低字节地址	数据高字节	数据低字节	CRC 码低字节	CRC 码高字节

当从站接收正确时, 从站回送:

字节	1	2	3	4	5	6	7	8
内容	ADR	0x06	寄存器高字节地址	寄存器低字节地址	数据高字节	数据低字节	CRC 码低字节	CRC 码高字节

当从站接收错误时, 从站回送:

字节	1	2	3	4	5
内容	ADR	0x86	异常码	CRC 低字节	CRC 高字节

第 1 字节 ADR: 从站地址码 (=001~254)
 第 2 字节 0x86: 写寄存器值出错功能码
 第 3 字节 异常码: 见 6.2.4 小节
 第 4、5 字节: 从字节 1 到 3 的 CRC16 校验码

0x10 写多个寄存器值

主站发送:

字节	1	2	3	4	5	6	7
内容	ADR	0x10	起始寄存器高字节地址	起始寄存器低字节地址	寄存器数量高字节	寄存器数量低字节	数据字节总数

字节	8,9	10,11	N,N+1	N+2	N+3
内容	寄存器数据 1	寄存器数据 2	寄存器数据 M	CRC 码低字节	CRC 码高字节

当从站接收正确时，从站回送：

字节	1	2	3	4	5	6	7	8
内容	ADR	0x10	寄存器高字节地址	寄存器低字节地址	寄存器数量高字节	寄存器数量低字节	CRC 码低字节	CRC 码高字节

当从站接收错误时，从站回送：

字节	1	2	3	4	5
内容	ADR	0x90	异常码	CRC 低字节	CRC 高字节

第 1 字节 ADR： 从站地址码（=001~254）

第 2 字节 0x90： 写寄存器值出错

第 3 字节 异常码：见 6.2.4 小节

第 4、5 字节：从字节 1 到 3 的 CRC16 校验码

错误异常码

1. MODBUS 异常码

表 6.1 MODBUS 异常码表

异常码	含义
0x01	非法功能码
0x02	非法数据地址
0x03	非法数据值
0x04	从站设备故障
0x05	请求已被确认，但需要较长时间来处理请求
0x06	从设备忙
0x08	存储奇偶性差错
0x0A	不可用的网关
0x0B	网关目标设备响应失败

3. 扩展异常码

表 6.2 扩展异常码表

异常码	含义
0x40	禁止操作
0x60	尚未学习电机相序
0xff	未定义错误

MODBUS寄存器定义

设备描述信息寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x0000	设备标识		0x03	

0x0001	设备版本号		0x03	高字节为主版本号，低字节为副版本号。
0x0002 0x0009	设备名称		0x03	以'\0'结束的字符串
0x000A	PWM 分辨率的倒数		0x03	
0x000B	PWM 频率		0x03	单位为 Hz
0x000C	最大输出电流		0x03	数值乘以 0.01 为电流值，单位为A。
0x000D	电流分辨率		0x03	单位为 mA
0x000E	保留		0x03	
0x000F	保留		0x03	

实时状态寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x0020	实时PWM	0~1000	0x03	数值乘以 0.1%为占空比
0x0021	实时电流	0~3500	0x03	数值乘以 0.01 为电流值，单位为A。
0x0022	实时换向频率(转速)	-32768~ 32767	0x03	当 0x0035 寄存器为 1 时，数值即为换向频率；当 0x0035 寄存器为 0 时，数值乘以 0.1 为换向频率；单位为 Hz；换向频率除以电机极个数再乘以 20 为电机转速，单位 RPM。
0x0023	位置控制完成状态	0, 1	0x03	0: 未完成 1: 完成
0x0024	电机实时位置高半字	-2147483648 ~ 2147483647	0x03	电机换向脉冲数
0x0025	电机实时位置低半字			
0x0026	剩余完成时间高半字	0~ 4294967295	0x03	单位为ms
0x0027	剩余完成时间低半字			
0x0028	IN1 电压	0~10000	0x03	单位为 mV
0x0029	IN2 电压	0~10000	0x03	单位为 mV
0x002a	IN3 电压	0~10000	0x03	单位为 mV
0x002b	差分电压	-3300~3300	0x03	单位为 mV
0x002c	SQ1 电平	0,1	0x03	0: 低电平 1: 高电平
0x002d	SQ2 电平	0,1	0x03	0: 低电平 1: 高电平
0x002e	IN1 输入占空比	0~1000	0x03	数值乘以 0.1%为占空比
0x002f	IN1 输入频率	0~100000	0x03	单位为 Hz
0x0030	IN1 输入脉冲高半字	-2147483648 ~ 2147483647	0x03	输入脉冲个数
0x0031	IN1 输入脉冲低半字			

0x0032	堵转状态	0, 1, 2	0x03	0: 未堵转 1: 正转堵转停止 2: 反转堵转停止
0x0033	错误状态	0~9	0x03	0: 无错误 1: 尚未学习 2: 堵转停止 3: 霍尔错误 4: 达不到目标速度 5: 线圈错误(本款不支持) 6: 过流关断 7: 过热关断 8: 过压关断 9: 欠压关断
0x0034	电机转速	0~65535	0x03	当 0x0035 寄存器为 1 时, 数值乘以 10 为转速; 当 0x0035 寄存器为 0 时, 数值即为转速; 单位为 RPM。 注: 需先通过 0x0073 和 0x0074 寄存器配置正确的电机极个数和减速比, 读取的转速才正确。
0x0035	转速是否需要乘以 10	0, 1	0x03	0: 数值即转速; 1: 数值乘以 10 为转速;
0x0037	内部(驱动电路)温度	-400~1250	0x03	数值乘以 0.1℃为温度
0x0038	电源电压	0~700	0x03	数值乘以 0.1V 为电压
0x0039	控制方式	0, 1	0x03	0: 本地控制 1: 通讯控制
0x003a	母线电流	0~3500	0x03	数值乘以 0.01A 为母线电流

速度控制寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x0040	停止	0, 1, 2	0x06	0: 正常停止 1: 紧急制动 2: 自由停止
0x0041	保留(力矩控制)	-3000~3000	0x06	数值乘以 0.01 为目标电流, 适合于电机堵转张力控制。
0x0042	设定占空比	-1000~1000	0x06	数值乘以 0.1%为目标占空比
0x0043	设定速度闭环控制目标速度(换向频率)	-32768~32767	0x06	数值乘以 0.1 为目标换向频率, 单位为 Hz
0x0044	设定位置闭环控制行走速度(换向频率)	0~32767	0x06	数值乘以 0.1 为目标换向频率, 单位为 Hz
0x0045	设定位置闭环控制类型	0, 1	0x06	0: 绝对位置 1: 相对位置

0x0046	设定位置闭环控制目标 位置高半字	-2147483648~ 2147483647	0x06	如果为绝对位置可任意时刻 改变目标位置；如果为相对位 置则要等待上一次位置控制 完成才可进行下一次操作
0x0047	设定位置闭环控制目标 位置低半字			
0x0048 0x004F	保留		不能访问	
0x0050	占空比调速加速缓冲时 间	0~255	0x03 0x06 0x10	数值乘以0.1 为输出占比空由 0 增加到 100.0%所需时间
0x0051	占空比调速减速缓冲时 间	0~255	0x03 0x06 0x10	数值乘以0.1 为输出占比空由 100.0%减小到 0 所需时间
0x0052	速度闭环控制、位置闭环 控制加速加速度	0~66635	0x03 0x06 0x10	数值乘以0.1 为换向频率增大 速度，单位为 Hz/s
0x0053	速度闭环控制、位置闭环 控制减速加速度	0~66635	0x03 0x06 0x10	数值乘以0.1 为换向频率减小 速度，单位为 Hz/s

注：当 0x0077 寄存器为 1 时，0x0043、0x0044、0x0052、0x0053 寄存器涉及的换向频率为寄存器的值，无须乘以 0.1。

电机控制参数配置寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x0060	上电时默认占空比调速加 速缓冲时间	0~255	0x03 0x06 0x10	数值乘以 0.1 为输出占比空 由 0 增加到 100.0%所需时间
0x0061	上电时默认占空比调速减 速缓冲时间	0~255	0x03 0x06 0x10	数值乘以 0.1 为输出占比空 由 100.0%减小到 0 所需时间
0x0062	速度闭环控制、位置闭环 控制最大加速加速度	0~66635	0x03 0x06 0x10	数值乘以 0.1 为换向频率最 大增大速度，单位为Hz/s
0x0063	上电时默认速度闭环/位 置闭环控制加速加速度	0~66635	0x03 0x06 0x10	数值乘以 0.1 为换向频率增 大速度，单位为Hz/s
0x0064	速度闭环控制、位置闭环 控制最大减速加速度	0~66635	0x03 0x06 0x10	数值乘以 0.1 为换向频率最 大减小速度，单位为Hz/s
0x0065	上电时默认速度闭环/位 置闭环控制减速加速度	0~66635	0x03 0x06 0x10	数值乘以 0.1 为换向频率减 小速度，单位为Hz/s
0x0066	速度闭环控制、位置闭环 控制最大速度(换向频率)	0~32767	0x03 0x06 0x10	数值乘以 0.1 为换向频率， 单位为 Hz
0x0067	上电时速度闭环控制/位 置闭环控制默认速度(换 向频率)	0~32767	0x03 0x06 0x10	数值乘以 0.1 为换向频率， 单位为 Hz
0x0068	保留	0	0x03 0x06 0x10	
0x0069	位置控制算法	0, 1, 2	0x03 0x06 0x10	0: 水平定位控制 1: 水平滑定位控制 2: 竖直定位控制
0x006a	电机额定电流	0~3000	0x03 0x06 0x10	数值乘以 0.01 为电流值，单 位为A。

0x006b	电机最大负载电流	0~3000	0x03 0x06 0x10	数值乘以 0.01 为电流值, 单位为A。
0x006c	电机最大制动电流	0~600	0x03 0x06 0x10	数值乘以 0.01 为电流值, 单位为A。
0x006d 0x006f	电机相序数据 6 字节	1~6	0x03 0x06 0x10	只有电机学习状态为未学习才能进行写操作, 否则写操作将被忽略
0x0070	速度闭环控制算法	0, 1, 2	0x03 0x06 0x10	0: 速度闭环控制 1: 时间-位置闭环控制 2: 时间-位置速率控制
0x0071	位置闭环控制允许误差	0~65535	0x03 0x06 0x10	
0x0072	位置闭环控制超调后修正	0, 1	0x03 0x06 0x10	0: 不进行修正 1: 进行修正
0x0073	电机极个数	0~65535	0x03 0x06 0x10	电机极对数乘以 2
0x0074	电机减速比	0~65535	0x03 0x06 0x10	数值乘以 0.1 为电机减速比
0x0075	电机学习状态	0, 1	0x03 0x06 0x10	0: 未学习 1: 已学习 (只能通过电机学习操作才能将未学习状态改为已学习状态, 不能直接写 1)
0x0076	禁用电机相序学习功能	0, 1	0x03 0x06 0x10	0: 未禁用 1: 禁用
0x0077	速度设定值×10	0, 1	0x03 0x06 0x10	0: 禁用 1: 启用
0x0078	常态自锁电流	0~3000	0x03 0x06 0x10	数值乘以 0.01 为电流值, 单位为A。 注意: 须小于电机额定电流的 1/2, 否则可能电机发热烧坏电机
0x0079	短时间最大输出电流为最大负载电流的倍数	0, 100~200	0x03 0x06 0x10	0: 禁用倍流 其它: 数值乘以 0.01 倍
0x007a	允许倍流输出时间	0~999	0x03 0x06 0x10	0: 禁用倍流 其它: 数值乘以 0.1 秒

注: 当 0x0077 寄存器为 1 时, 0x0062~0x0067 寄存器涉及的换向频率为寄存器的值, 无须乘以 0.1。

系统参数配置寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x0080	限位触发极性	0,1,2,3,4	0x03 0x06 0x10	0: 低电平触发 1: 高电平触发 2: 下降沿触发 3: 上升沿触发 4: 禁用限位功能

0x0081	数字信号极性	0,1,2,3	0x03 0x06 0x10	0: 低电平触发 1: 高电平触发 2: 下降沿触发 3: 上升沿触发
0x0082	电位器用法	0,1,2	0x03 0x06 0x10	0: 单电位器 1: 双电位器独立 2: 双电位器协同
0x0083	脉冲信号类型	0,1,2	0x03 0x06 0x10	0: PWM 1: 频率 2: 脉冲
0x0084	模拟信号类型	0,1,2,3	0x03 0x06 0x10	0: 单端信号 1: 差分信号 2: 双单端信号独立 3: 双单端信号协同
0x0085	逻辑电平类型	0,1,2,3	0x03 0x06 0x10	0: 开关量 1: 0/3.3V 2: 0/5V 3: 0/12V 或 0/24V
0x0086	电位器最小值	0~10000	0x03 0x06 0x10	单位为 mV
0x0087	电位器最大值	0~10000	0x03 0x06 0x10	单位为 mV
0x0088	模拟量范围最小值	0~10000	0x03 0x06 0x10	单位为 mV
0x0089	模拟量范围最大值	0~10000	0x03 0x06 0x10	单位为 mV
0x008a	逻辑电平阈值	0~10000	0x03 0x06 0x10	单位为 mV
0x008b	电位比较死区	0~10000	0x03 0x06 0x10	单位为 mV
0x008c	脉冲信号倍率四字节浮点 型高半字		0x03 0x06 0x10	
0x008d	脉冲信号倍率四字节浮点 型低半字			
0x008e	堵转停止时间	0~255	0x03 0x06 0x10	数值乘以 0.1 为堵转停止时间, 单位为s
0x008f	保留			
0x0090	波特率高半字	9600~ 115200	0x03 0x06 0x10	
0x0091	波特率低半字			
0x0092	校验方式	0,1,2	0x03 0x06 0x10	0: 无校验+2 停止位 1: 奇校验+1 停止位 2: 偶校验+1 停止位

0x0093	485 控制默认调速方式	0, 1, 2	0x03 0x06 0x10	0: 占空比 1: 速度闭环控制 2: 位置闭环控制
0x0094	485 控制时禁止参数配置	0, 1	0x03 0x06 0x10	0: 不禁止 1: 禁止
0x0095	通讯中断停止时间	0~255	0x03 0x06 0x10	数值乘以 0.1 为通讯中断停止时间, 单位为s
0x0096	模拟信号调整系数k 四字 节浮点型高半字		0x03 0x06 0x10	不能小于 0
0x0097	模拟信号调整系数k 四字 节浮点型低半字			
0x0098	模拟信号调整系数b	0~65535	0x03 0x06 0x10	单位为 mV
0x0099	禁用报警	0, 1	0x03 0x06 0x10	0: 不禁用 1: 禁用
0x009A	保留			
0x009B	保留			
0x009C	指定 485 从站地址	0~127	0x03 0x06 0x10	0: 通讯控制方式时由拨码开关 1~7 位设定从站地址, 数字/模拟信号控制方式时从站地址固定为 0x01; 1~127: 指定通讯控制方式时的从站地址, 拨码开关设定的地址失效。
0x009C	数字/模拟信号控制方式 时是否指定485 从站地址	0,1	0x03 0x06 0x10	0: 数字/模拟信号控制时 485 从站地址固定为 0x01; 1: 数字/模拟信号控制时 485 从站地址由0x009c 寄存器指定。

往复位置控制参数

寄存器地址	描述	取值范围	支持功能码	备注
0x00a0	复位模式	0,1,2,3,4	0x03 0x06 0x10	0: 不复位 1: SQ2 复位 2: SQ1 复位 3: SQ2 复位并细调 4: SQ1 复位并细调
0x00a1	是否启用复位细调	0, 1	0x03 0x06 0x10	0: 禁用 1: 启用
0x00a2	总行程高半字	-2147483648	0x03 0x06 0x10	
0x00a3	总行程低半字	~2147483647		
0x00a4	复位粗调速度	0~65535	0x03 0x06 0x10	乘以 0.1 为换向频率

0x00a5	复位细调速度	0~65535	0x03 0x06 0x10	乘以 0.1 为换向频率
0x00a6	到端点后最终速度	0~65535	0x03 0x06 0x10	乘以 0.1 为换向频率
0x00a7	要忽略的信号变化量	0~1000	0x03 0x06 0x10	数值乘以 0.1%为要忽略的输入信号变化量比例；用以解决电位器、模拟信号、占空比或频率信号波动问题
0x00a8	限位后是否重新复位	0, 1	0x03 0x06 0x10	0: 否 1: 是 用以解决机械轮子打滑造成的行程误差问题
0x00a9	复位时转矩	0~1500	0x03 0x06 0x10	0: 最大转矩 1: 配置的电流对应的转矩
0x00aa	复位测试	0,1,2,3,4	0x03 0x06 0x10	0: 非复位状态 1: 取消复位 2: SQ1 复位 3: SQ2 复位 4: 测量行程

预设速度寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x00b0	工作模式	0,1,2,3	0x03 0x06 0x10	0: 占空比 1: 力矩 2: 速度闭环 3: 位置闭环
0x00b1	控制方式	0, 1	0x03 0x06 0x10	0: 双触点/逻辑电平控制 1: 单触点/逻辑电平控制
0x00b2	正转速度	占空比方式: 0~1000 力矩方式: 0~3000 速度位置闭环: 0~65536	0x03 0x06 0x10	数值乘以 0.1%为占空比; 数值乘以 0.01 为力矩; 数值乘以 0.1 为换向频率;
0x00b3	反转速度	占空比方式: 0~1000 力矩方式: 0~3000 速度位置闭环: 0~65536	0x03 0x06 0x10	数值乘以 0.1%为占空比; 数值乘以 0.01 为力矩; 数值乘以 0.1 为换向频率;

注：当 0x0077 寄存器为 1 时，0x0062~0x0067 寄存器涉及的换向频率为寄存器的值，无须乘以 0.1。

闭环控制PID参数配置寄存器

寄存器地址	描述	取值范围	支持功能码	备注
-------	----	------	-------	----

0x00ba	位置自锁P 系数四字节浮点 型高半字	建议 0.1~100	0x03 0x06 0x10	
0x00bb	位置自锁P 系数四字节浮点 型低半字			
0x00bc	位置自锁I 系数四字节浮点 型高半字	建议 0.001~1	0x03 0x06 0x10	
0x00bd	位置自锁I 系数四字节浮点 型低半字			
0x00be	位置自锁D 系数四字节浮点 型高半字	建议 0.001~1	0x03 0x06 0x10	
0x00bf	位置自锁D 系数四字节浮点 型低半字			
0x00c0	速度闭环控制P 系数四字节 浮点型高半字	建议 0.001~1	0x03 0x06 0x10	
0x00c1	速度闭环控制P 系数四字节 浮点型低半字			
0x00c2	速度闭环控制I 系数四字节 浮点型高半字	建议 0.001~1	0x03 0x06 0x10	
0x00c3	速度闭环控制I 系数四字节 浮点型低半字			
0x00c4	速度闭环控制D 系数四字节 浮点型高半字	建议 0.001~1	0x03 0x06 0x10	
0x00c5	速度闭环控制D 系数四字节 浮点型低半字			
0x00c6	位置闭环控制P 系数四字节 浮点型高半字	建议 0.1~100	0x03 0x06 0x10	
0x00c7	位置闭环控制P 系数四字节 浮点型低半字			
0x00c8	位置闭环控制I 系数四字节 浮点型高半字	建议 0.001~1	0x03 0x06 0x10	
0x00c9	位置闭环控制I 系数四字节 浮点型低半字			
0x00ca	位置闭环控制D 系数四字节 浮点型高半字	建议 0.001~1	0x03 0x06 0x10	
0x00cb	位置闭环控制D 系数四字节 浮点型高半字			

电机学习寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x00e0	保留	0	0x03	请勿操作该寄存器

0x00e1	学习命令	0,1	0x03 0x06 0x10	0: 未学习 1: 开始电机学习/学习中
0x00e2	学习状态	0, 1, 2, 3, 4	0x03	0: 准备就绪 1: 正在学习 2: 正在停止 3: 学习完毕 4: 学习失败
0x00e3	学习进度		0x03	子过程号
0x00e4	学习结果数据字节数		0x03	
0x00e5 0x00ef	学习结果数据		0x03	

安全保护寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x0100	过热关断触发温度	-40~125	0x03 0x06 0x10	温度达到该值后关断输出
0x0101	禁用倍流触发温度	-40~125	0x03 0x06 0x10	温度达到该值后禁用倍流输出
0x0102	过压关断触发电压	80~660	0x03 0x06 0x10	数值乘以 0.1V 为电压； 电压超过该值后关断输出
0x0103	欠压关断触发电压	80~600	0x03 0x06 0x10	数值乘以 0.1V 为电压； 低于该值后关断输出
0x0104	过流关断触发电流	0~5500	0x03 0x06 0x10	数值乘以 0.01A 为电流； 电流峰值达到该值后关断输出
0x0105	霍尔错误屏蔽时间	0~100	0x03 0x06 0x10	单位 ms；当霍尔错误状态 达到该时长后关断输出
0x0106	启用自动调节电流环系数	0, 1	0x03 0x06 0x10	0: 禁用 1: 启用；当启动电流上升 过快时减小PID 系数
0x0107	保留		0x03 0x06 0x10	
0x0108	启用当温度低于过热保护 触发值后自动清除报警	0, 1	0x03 0x06 0x10	0: 禁用 1: 启用
0x0109	保留		0x03 0x06 0x10	
0x010a	温度校正系数K（倍数）	9500~10500	0x03 0x06 0x10	数值乘以 0.0001 倍
0x010b	温度校正系数B（截距）	-100~100	0x03 0x06 0x10	数值乘以 0.1℃
0x010c	电压校正系数K（倍数）	9700~10300	0x03 0x06 0x10	数值乘以 0.0001 倍

0x010d	电压校正系数B（截距）	-10~10	0x03 0x06 0x10	数值乘以 0.1V
--------	-------------	--------	-------------------	-----------

CAN参数配置寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x0120	CAN 通讯模式	0,1	0x03 0x06 0x10	0: 485/CAN 模式 1: CANopen 模式
0x0121	CAN 节点 ID	0~127	0x03 0x06 0x10	0: CAN 通讯控制时节点 ID 由拨码开关指定，数字/模拟信号控制时节点ID 为 0x01； 1~127: 指定 CAN 通讯控制时的节点 ID，拨码开关设定地址无效。
0x0122	CAN 波特率	0~7	0x03 0x06 0x10	0: 10kbps 1: 20kbps 2: 50kbps 3: 125kbps 4: 250kbps 5: 500kbps 6: 800kbps 7: 1Mbps
0x0123 0x0127	保留			
0x0128	CANopen 自启动	0,1	0x03 0x06 0x10	0: CANopen 模式时，上电后最终进入 Pre-Operational 状态，需客户端通过 NMT 服务使驱动器进入 Operational 状态； 1: CANopen 模式时，上电后最终进入 Operational 状态。
0x0129	CANOpen 心跳周期	0~65535	0x03 0x06 0x10	0: 禁用心跳包； 1: 驱动器以设定的周期发送心跳包，单位ms。

对象字典操作寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x0200	命令字节	0x23, 0x2B, 0x27, 0x2F, 0x40, 0x60, 0x80	0x03 0x06 0x10	0x06 和 0x10 功能码支持命令 0x2F: 写单字节数据 0x2B: 写双字节数据 0x27: 写三字节数据 0x23: 写四字节数据 0x40: 读数据 0x03 功能读取返回命令 0x4F: 有效数据为一个字节 0x4B: 有效数据为两个字节 0x47: 有效数据为三个字节 0x43: 有效数据为四个字节 0x60: 写入数据传送成功 0x80: 写入数据传送中止
0x0201	索引号		0x03 0x06 0x10	对象字典索引号
0x0202	子索引号		0x03 0x06 0x10	对象字典子索引号
0x0204	数据高半字	1	0x03 0x06 0x10	数据第 3、4 字节
0x0205	数据低半字	1	0x03 0x06 0x10	数据第 1、2 字节
0x0206	应用命令	1	0x06	应用以上寄存器写入值对应的命令；若由 0x10 功能码操作 0x0200~0x0205 寄存器，则自动应用命令，无须操作此寄存器。

注：485 通讯方式下可通过以上 Modbus 寄存器访问 CANopen 对象字典。在写入命令、索引号、子索引号和数据并应用命令后，通过读取以上寄存器可获得返回状态和数据。数据长度大于 4 个字节的对象无法通过此方式访问。

配置参数存储寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x0160	存储电机配置参数	1	0x06	
0x0180	存储系统配置参数	1	0x06	
0x01a0	存储往复控制参数	1	0x06	
0x01b0	存储预设速度参数	1	0x06	
0x01c0	存储闭环调速PID 参数	1	0x06	
0x01d0	存储安全保护参数	1	0x06	
0x01e0	存储 CAN 配置参数	1	0x06	
0x01e8	存储 CANopen 参数	1	0x06	
0x01f0	存储用户过程数据	1	0x06	

注：通过 0x10 功能码配置的参数可直接存储到驱动器中，而通过 0x06 功能码配置的参数，则需要通过以上寄存器才能存储到存储器。

程序操作寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x00f0	虚拟机版本		0x03	
0x00f1	程序空间大小		0x03	
0x00f2	运行状态		0x03	
0x00f3	位置控制完成状态	0,1	0x03	0：未完成 1：完成
0x00f4 0x00f9	保留			
0x00fa	设备地址		0x03 0x06	
0x00fb	是否自动运行		0x03 0x06	

IO配置寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x00d0	IO1 方向	0,1,2	0x06	0: 浮空输入; 2: 输出; 1: 上拉下拉输入
0x00d1	IO1 电平	0,1	0x06	0: 低电平/下拉; 1: 高电平/上拉
0x00d2	IO2 方向	0,1,2	0x06	0: 浮空输入; 2: 输出; 1: 上拉下拉输入
0x00d3	IO2 电平	0,1	0x06	0: 低电平/下拉; 1: 高电平/上拉
0x00d4	IO3 方向	0,1,2	0x06	0: 浮空输入; 2: 输出; 1: 上拉下拉输入
0x00d5	IO3 电平	0,1	0x06	0: 低电平/下拉; 1: 高电平/上拉

外设操作相关寄存器

寄存器地址	描述	取值范围	支持功能码	备注
0x7000	3.3V 输出	0,1	0x06	0: 禁用 1: 使能
0x7001	报警	0,1	0x06	0: 禁用 1: 使能
0x7002	输入类型	0,1	0x06	0: 模拟 1: 数字
0x7003	输入脉冲方向	0,1	0x06	0: 减少 1: 增加
0x7004	清零输入脉冲	0,1,2	0x06	0: 关闭自动清除零 1: 手动清除零 2: 启动自动清除零
0x7005 0x7009	保留			
0x700a	清零位置计数	0,1	0x06	0: 无动作 1: 清零位置计数
0x700b 0x700f	保留			
0x7010 0x7011	读取输入脉冲		0x03	如果 0x7004 写了 2, 则读取后 此寄存器将自动清零
0x7012	读取输入占空比	0-1000	0x03	占空比乘以 1000
0x7013	读取输入频率		0x03	
0x7014	读取霍尔输入状态		0x03	

0x7015 0x701f	保留			
0x7020	IO1 电位器百分比	0-1000	0x03	百分比乘以 1000
0x7021	IO1 电压百分比	0-1000	0x03	百分比乘以 1000
0x7022	IO1 电平		0x03	0: 低电平 1: 高电平
0x7023	IO1 边沿		0x03 0x06	0: 下降 1: 上升 2: 未触发 写入任意: 清除触发
0x7024	保留			
0x7025	IO2 电位器百分比	0-1000	0x03	百分比乘以 1000
0x7026	IO2 电压百分比	0-1000	0x03	百分比乘以 1000
0x7027	IO2 电平		0x03	0: 低电平 1: 高电平
0x7028	IO2 边沿		0x03 0x06	0: 下降 1: 上升 2: 未触发 写入任意: 清除触发
0x7029	保留			
0x702a	IO3 电位器百分比	0-1000	0x03	百分比乘以 1000
0x702b	IO3 电压百分比	0-1000	0x03	百分比乘以 1000
0x702c	IO3 电平		0x03	0: 低电平 1: 高电平
0x702d	IO3 边沿		0x03 0x06	0: 下降 1: 上升 2: 未触发 写入任意: 清除触发
0x702e	保留			
0x7040	定时器 0 计数器高字节	0-2000000000	0x06	
0x7041	定时器 0 计数器低字节			
0x7042	定时器 0 控制寄存器		0x06	0: 禁止 1: 使能单次触发 (自动清除定时触发标志) 0x81: 使能单次触发 (手动清除定时触发标志) 2: 使能周期性触发
0x7043	定时器 0 触发状态		0x03	0: 未触发 1: 触发
0x7044	定时器 1 计数器高字节	0-2000000000	0x06	

0x7045	定时器 1 计数器低字节			
0x7046	定时器 1 控制寄存器		0x06	0: 禁止 1: 使能单次触发（自动清除定时触发标志） 0x81: 使能单次触发（手动清除定时触发标志） 2: 使能周期性触发
0x7047	定时器 1 触发状态		0x03	0: 未触发 1: 触发
0x7048	定时器 2 计数器高字节	0-2000000000	0x06	
0x7049	定时器 2 计数器低字节			
0x704a	定时器 2 控制寄存器		0x06	0: 禁止 1: 使能单次触发（自动清除定时触发标志） 0x81: 使能单次触发（手动清除定时触发标志） 2: 使能周期性触发
0x704b	定时器 2 触发状态		0x03	0: 未触发 1: 触发
0x704c	定时器 3 计数器高字节	0-2000000000	0x06	
0x704d	定时器 3 计数器低字节			
0x704e	定时器 3 控制寄存器		0x06	0: 禁止 1: 使能单次触发（自动清除定时触发标志） 0x81: 使能单次触发（手动清除定时触发标志） 2: 使能周期性触发
0x704f	定时器 3 触发状态		0x03	0: 未触发 1: 触发
0x7060	系统节拍高字节	长整型	0x03	
0x7061	系统节拍低字节			
0x7062	上一次过程ID		0x03	
0x7063	当前过程 ID		0x03	
0x7100 0x711F	通用寄存器		0x03 0x06	
0x8000	程序代码/调试日志		0x03	调试日志

CAN通讯协议

本驱动器支持 RS485/CAN 模式和 CANopen 模式通讯，通过 0x0120 寄存器来指定通讯模式。若为 RS485/CAN 通讯模式，当驱动器上电后检测到的第一帧数据为 Modbus 通讯帧，则驱动器使用 485 通讯方式通讯。若驱动器检测到的第一帧数据为 CAN 通讯帧，则驱动器使用 CAN 通讯方式通讯。若为 CANopen 通讯模式，则驱动器只能使用 CAN 通讯方式通讯。

在CAN通讯方式下，驱动器支持多节点通讯。通过拨码开关或寄存器/对象字典可配置 127 个节点ID，客户端可以是单片机、PLC或PC机等。关于节点ID的配置见 2.1.5 小节。

CAN通讯参数

本驱动器支持 CAN 标准帧，驱动器采用 11 位 CAN 标识符的第 0~6 位作为 CAN 节点 ID，第 7~10 位作为功能码。驱动器不支持扩展帧和远程帧。

CAN节点ID可通过拨码开关、0x0121 寄存器或对象字典 0x2201 索引配置。当 0x0121 寄存器和 0x2201 索引数据为默认值 0 时，通讯控制方式（拨码开关第 8 位为ON）的CAN节点ID由拨码开关设定（节点ID设定方法见 2.1.5 小节），数字/模拟信号控制方式（拨码开关第 8 位为OFF）的CAN节点ID为 0x01；当 0x0121 寄存器或 0x2201 索引数据为 1~127 范围内的值时，通讯控制方式和数字/模拟信号控制方式的CAN节点ID均由为该值。

CAN 波特率可通过 0x0122 寄存器或对象字典 0x2202 索引配置。默认值为 500kbps，可配置范围为 10kbps~1Mbps。

我们可通过配置 0x0120 寄存器值为 1 来启用 CANOpen 协议；通过配置 0x0128 寄存器值为 1 来启用 CANopen 自启动（驱动器上电节点初始化后将直接进入 Operational 状态）；通过 0x0129 寄存器或基本对象字典的 0x1017 索引来配置心跳周期。

当拨码开关第 1~8 位全为 ON 时，通讯参数为默认通讯参数，CAN 波特率为 500kbps，CAN 节点 ID 为 0x01，此时不支持 CANopen 通讯协议（即使 0x0120 寄存器值配置为 1）。

CAN消息语法

1. 主站/客户端发送消息格式

在RS485/CAN通讯模式或CANopen通讯模式，我们可通过如下CAN消息格式（对应于CANopen的SDO加速传送报文）来访问对象字典，对象字典定义见 7.6 节。

CAN 标识符	数据字节 0	数据字节 1-2	数据字节 3	数据字节 4-7
0x600+节点 ID	命令	索引号	子索引号	数据

CAN 标识符第 0~6 位：节点 ID（0x01~0x7F），每一个 CAN 设备的节点 ID 应唯一

数据字节 0：命令字节（写 1~4 字节数据的命令依次为 0x2F、0x2B、0x27、0x23，读数据命令为 0x40）

数据字节 1-2：对象字典索引号，低字节在前

数据字节 3：对象字典子索引号

数据字节 4-7：要写入的数据，低字节在前，对于每个无效字节，可用 0x00 填充
命令字节与数据字节 4-7 中有效数据字节对应关系如下：

命令	功能	数据字节 4-7 中有效字节
0x2F	写单字节	前 1 字节
0x2B	写双字节	前 2 字节
0x27	写三字节	前 3 字节
0x23	写四字节	全有效
0x40	读取数据	均无效

注：在 RS485/CAN 通讯模式，无法访问数据长度大于 4 字节的对象，若要访此类对象，须在 CANopen 通讯模式下进行，相应语法请参阅 CANopen 协议相关文档。

2. 从站/服务器应答消息格式

相应 CAN 节点接收到消息后的应答格式如下：

CAN 标识符	数据字节 0	数据字节 1-2	数据字节 3	数据字节 4-7
0x580+节点 ID	状态	索引号	子索引号	数据/中止代码

CAN 标识符第 0~6 位：节点 ID（0x01~0x7F）

数据字节 0：状态码

数据字节 1-2：对象字典索引号，低字节在前

数据字节 3：对象字典子索引号

数据字节 4-7：读取的数据或传送中止代码，低字节在前

状态码与数据字节 4-7 中有效数据对应关系如下：

状态码	描述	数据字节 4-7 中有效字节
0x4F	数据长度为 1 字节	前 1 字节
0x4B	数据长度为 2 字节	前 2 字节
0x47	数据长度为 3 字节	前 3 字节
0x43	数据长度为 4 字节	全有效
0x60	传送成功	均无效
0x80	传送中止	4 个字节为中止代码

传送中止代码

RS485/CAN 模式读写数据和 CANopen 模式 SDO 上传下载数据传送中止的代码如下：

中止代码	功能描述
0x05030000	触发位没有交替改变
0x05040000	SDO协议超时
0x05040001	非法或未知的Client/Server 命令字
0x05040002	无效的块大小（仅Block Transfer模式）
0x05040003	无效的序号（仅Block Transfer模式）
0x05030004	CRC错误（仅Block Transfer模式）

0x05030005	内存溢出
0x06010000	对象不支持访问
0x06010001	试图读只写对象
0x06010002	试图写只读对象
0x06020000	对象字典中对象不存在
0x06040041	对象不能够映射到PDO
0x06040042	映射的对象的数目和长度超出PDO长度
0x06040043	一般性参数不兼容
0x06040047	一般性设备内部不兼容
0x06060000	硬件错误导致对象访问失败
0x06060010	数据类型不匹配，服务参数长度不匹配
0x06060012	数据类型不匹配，服务参数长度太大
0x06060013	数据类型不匹配，服务参数长度太短
0x06090011	子索引不存在
0x06090030	超出参数的值范围(写访问时)
0x06090031	写入参数数值太大
0x06090032	写入参数数值太小
0x06090036	最大值小于最小值
0x08000000	一般性错误
0x08000020	数据不能传送或保存到应用
0x08000021	由于本地控制导致数据不能传送或保存到应用
0x08000022	由于当前设备状态导致数据不能传送或保存到应用
0x08000023	对象字典动态产生错误或对象字典不存在 (例如，通过文件生成对象字典，但由于文件损坏导致错误产生)

通过CAN访问MODBUS寄存器

在 CAN 通讯方式下，我们可通过 Modbus 寄存器地址与对象字典索引号的转换来实现通过访问对象字典对 Modbus 寄存器的访问。Modbus 寄存器地址与 CANopen 索引号的转换关系如下：

MODBUS 寄存器地址 (REG_ADDR)	对象字典索引号 (INDEX)	对象字典子索引号 (SUB-INDEX)
0x0000~0x0FFF	0x4000 + REG_ADDR	0
0x7000~0x7FFF	REG_ADDR - 0x2000	0

我们可通过 0x40 命令对 Modbus 寄存器进行读取操作。对于数据长度为 16 位的寄存器通过 0x2B 命令进行写入操作，对于数据长度为 32 位的寄存器（如：0x0046 设定目标位置寄存器）通过 0x23 命令进行写入操作。CAN 消息语法见 7.2 节。

驱动器对CANopen的支持

CANopen通讯对象支持情况

在 CAN 通讯方式 CANopen 模式下，驱动器对 CANopen 协议进行有限的支持，支持情况如下：

通讯对象	项目	支持情况
NMT（网络管理）	NMT 模块控制（NMT Module Control）	支持
	NMT 启动（NMT Boot-up）	支持
	NMT 心跳包（NMT Heartbeat）	支持
	NMT 节点 守卫（NMT Node Guarding）	不支持
SDO（服务数据对象）	支持 SDO 数量	1
	修改 SDO COB-ID	不支持
	加速传送（Expedited transfer）	支持
	分段传送（Segmented transfer）	支持
	块传送（Block transfer）	不支持
PDO（过程数据对象）	支持RPDO 数量	4
	支持TPDO 数量	4
	修改 PDO COB-ID	支持
	修改 PDO 映射参数	支持
	PDO 传输类型	同步：支持 SYNC 消息同步，不支持远程帧同步 异步：支持定时事件触发，不支持远程帧触发
	TPDO 事件时间	支持
	TPDO 禁止时间	最短有效时间 10 个单位 (10*100us=1ms)
	MPDO（multiplexor PDO）	不支持
SYNC（同步）	修改 SYNC COB-ID	支持
	SYNC 通信间隔时间	最短有效时间 1000us
	同步窗口长度	最短有效时间 1000us
Time Stamp（时间标记对象）		不支持
Emergency Object（应急指示对象）	修改EMCY COB-ID	不支持
	EMCY 禁止时间	最短有效时间 10 个单位 (10*100us=1ms)

CANopen 具体规范可参阅《CiA DS301 V4.02 - CANopen Application Layer and Communication Profile》。

网络管理（NMT）

在 CANopen 模式下，驱动器支持 CANopen NMT 服务的 NMT 控制模块服务、NMT 启动和 NMT 心跳包。

1. NMT 控制模块服务（NMT Module Control）

NMT-Master 可通过 NMT Module Control 报文配置驱动器的节点状态，NMT Module Control 消息无应答消息，NMT 消息格式如下：

CAN 标识符	数据字节 0	数据字节 1
0x000	命令	节点 ID

CAN 标识符：NMT Module Control 报文的 CAN 标识符为 0

数据字节 0：命令字

数据字节 1：要操作的设备的节点 ID

其中，命令字对应的作用如下：

命令字	NMT 服务
0x01	启动远程节点（Start Remote Node）
0x02	停止远程节点（Stop Remote Node）
0x80	进入预运行状态（Enter Pre-operational State）
0x81	重启节点（Reset Node）
0x82	重启通讯（Reset Communication）

2. NMT 启动（NMT Boot-up）

当驱动器上电后，驱动器会发布 Boot-up 报文通知 NMT-Master 节点它已经从由 Initialising 状态转换为 Pre_Operational 状态，Boot-up 报文格式如下：

CAN 标识符	数据字节 0
0x700+节点 ID	0

CAN 标识符第 0~6 位：驱动器自身的节点 ID（0x01~0x7F）

数据字节 0：单字节数据 0

3. NMT 心跳包（NMT Heartbeat）

当基本对象字典 0x1017 索引配置为非 0 时（单位：ms），驱动器将周期性地产生心跳报文(Heartbeat)传送它的状态给心跳包消费者，消息格式如下：

CAN 标识符	数据字节 0
0x700+节点 ID	状态

CAN 标识符第 0~6 位：驱动器自身的节点 ID（0x01~0x7F）

数据字节 0：驱动器的节点状态，节点状态可为以下值：

状态值	意义
0x00	启动（Boot-up）
0x04	停止（Stopped）
0x05	运行（Operational）
0x7F	预运行（Pre-operational）

服务数据对象（SDO）

SDO 用来访问设备的对象字典，访问者称作客户端（Client），被访问且提供所请求服务的 CANopen 设备称作服务器（Server）。

其中，加速传送（Expedited transfer）方式可传送不超过 4 个字节的数据，SDO加速传送的消息格式见 7.2 节。

超过 4 字节的数据可通过分段传送（Segmented transfer）方式来访问，由于此传送方式

对驱动器的使用作用不大，这里就不进行具体介绍。

过程数据对象（PDO）

1. PDO 概述

PDO 用于传输实时过程数据，PDO 按照生产-消费者模式传输数据，每个 PDO 通过单个 CAN 通讯帧来传输数据，通讯帧里的 8 个字节数据可全部用作过程数据，一个 PDO 至多可传送 8 个字节数据。通讯帧的 CAN 标识符由 PDO 类型（第 7~10 位）和设备节点 ID（第 0~6 位）组成。通过 PDO 类型来识别传送的哪一个 PDO，节点 ID 与 PDO 帧标识符中指定的节点 ID 一致的设备才会对接收到的 PDO 消息进行处理。PDO 标识符的格式如下：

CAN 标识符 7~10 位	CAN 标识符 0~6 位
PDO 类型	节点 ID

PDO 数据传送无协议控制，PDO 传送数据内容由 PDO 映射定义。

通过 TPDO 映射参数可将对象字典多个索引的（合计至多 8 个字节的）实时数据按照映射关系“打包”到一个 CAN 通讯帧，当特定事件触发（如：定时事件触发）或接收到设定数量的 SYNC 消息（由 PDO 通讯参数配置触发方式）后，驱动器通过 TPDO 将“打包”的数据传送给相应的 PDO 消费者。因此，我们可通过 TPDO 高效地传送驱动器的实时状态数据。

通过 RPDO 映射参数可将一个 CAN 通讯帧的至多 8 个字节按照映射关系解析拆分为多个数据并依次向对象字典多个索引写入相应数据。因此，我们可通过 RPDO 高效地向驱动器传送控制数据。

2. PDO 通讯参数和映射参数的设置

驱动器支持 4 个 TPDO 和 4 个 RPDO，其对应的通讯参数索引号、映射参数索引号和默认 CAN 标识符如下：

PDO 类型	默认 COB-ID (CAN 标识符)	PDO 通讯参数索引号	PDO 映射参数索引号
RPDO1	0x200+节点 ID	0x1400	0x1600
RPDO2	0x300+节点 ID	0x1401	0x1601
RPDO3	0x400+节点 ID	0x1402	0x1602
RPDO4	0x500+节点 ID	0x1403	0x1603
TPDO1	0x180+节点 ID	0x1800	0x1A00
TPDO2	0x280+节点 ID	0x1801	0x1A01
TPDO3	0x380+节点 ID	0x1802	0x1A02
TPDO4	0x480+节点 ID	0x1803	0x1A03

通过 SDO 访问对象字典每个 PDO 的通讯参数索引的子索引号 1 可修改该 PDO 的 CAN 标识符（COB-ID），若该 PDO 在正常使用，应先对该 PDO 的 COB-ID 的第 31 位进行或操作先停用该 PDO，再设置新的 COB-ID。通过 SDO 访问对象字典每个 PDO 通讯参数索引的子索引号 2 可设置该 PDO 的传输类型，PDO 传输类型的定义如下：

传输类型	触发PDO的条件 (B = both needed O = one or both)			PDO传输
	SYNC	RTR	Event	
0	B	-	B	同步，非循环

1-240	O	-	-	同步, 循环
241-251	-	-	-	Reserved
252	B	B	-	同步, 在RTR之后 (不支持)
253	-	O	-	异步, 在RTR之后 (不支持)
254	-	O	O	异步, 制造商特定事件
255	-	O	O	异步, 设备子协议特定事件
说明: < SYNC—接收到SYNC-object。 < RTR —接收到远程帧。 < Event—例如数值改变或者定时器中断。 < 传输类型为: 1到240时, 该数字代表两个PDO之间的SYNC对象的数目)。				

每个TPDO可通过PDO通讯参数索引的子索引号3指定禁止时间, 即定义两个连续PDO传输的最小间隔时间, 避免由于高优先级信息的数据量太大, 始终占据总线, 而使其它优先级较低的数据无力竞争总线的问题。禁止时间由 16 位无符号整数定义, 单位 100us。

每个 TPDO 可通过通讯参数索引的子索引号 5 指定事件定时周期, 当超过定时时间后, 此 PDO 传输可以被触发 (不需要触发位)。事件定时周期由 16 位无符号整数定义, 单位 1ms。

若要修改 PDO 映射, 应先通过 SDO 向 PDO 映射参数索引的子索引号 0 写 0, 再根据需要通过子索引号 1~8 映射项 (Object Mapped) 设置映射关系。每一个映射项为 32 位无符号数据, 其中, 位 16~31 为索引号, 位 8~15 为子索引号, 位 0~7 为数据位数 (如: 长整型数据的位数为 32), 映射项的数据格式如下:

映射项 16~31 位	映射项 8~15 位	映射项 0~7 位
索引号	子索引号	数据位数

一个 PDO 的所有映射项映射数据长度之和不得超过 8 个字节。设置好映射项后, 再向映射参数索引的子索引号 0 写入映射项的数量。

若要存储 PDO 通讯参数和映射参数, 还须通过 SDO 向对象字典 0x1010 索引的子索引号 2 写入数值 0x65766173。

3. PDO 映射示例

示例一: 通过 RPDO 映射电机控制相关索引, 实现通过 RPDO1 对电机进行占空比调速、闭环调速、位置控制和制动控制操作。

RPDO1 的通讯参数我们可以使用默认参数, 那么 RPDO1 的 COB-ID 为 0x200+节点 ID; 而 RPDO1 的映射参数索引号为 0x1600, 我们先通过对象字典索引号 0x1600 的子索引号 0 写入单字节数据 0 使映射无效, 然后对 1~8 子索引进行如下设置:

索引号	子索引号	数据	映射内容
0x1600	1	0x20000008	索引号: 0x2000 子索引号: 0 数据位数: 8 对象作用: 设定电机控制类型
	2	0x20010010	索引号: 0x2001 子索引号: 0 数据位数: 16 对象作用: 设定电机控制量
	3	0x20020008	索引号: 0x2002 子索引号: 0 数据位数: 8 对象作用: 设定位置类型
	4	0x20030020	索引号: 0x2003 子索引号: 0 数据位数: 32 对象作用: 设目标位置

以上 4 个映射项的数据长度之和已经达到 8 个字节, 我们已不能再对此 PDO 进行

更多数据对象的映射了，我们对索引号 0x1600 的子索引号 0 写入映射项数量 4 使映射生效。若需要掉电保存该映射参数，我们可通过对 0x1010 索引的子索引号 2 写入数值 0x65766173 将相关参数储存到驱动器中。

我们配置好 RPDO1 的映射参数后，接下来我们来验证映射的 RPDO1 的功能，假设已配置驱动器的节点 ID 为 1（可通过配置 0x0120 和 0x0121 寄存器的值分别为 1 和 0，并且拨码开关第 8 位拨到 ON，第 1-7 位拨到 OFF 实现），那么 RPDO1 的 COB-ID 则为 0x201。

当驱动器节点状态为 Operational 时，我们可以通过向驱动器发送如下格式的 CAN 消息实现设定电机以 500.0Hz 的换向频率转动到目标位置 500。

CAN 标识符	数据字节 0~7（16 进制，低字节在前）
0x201	03 88 13 00 F4 01 00 00

注：

- 1) 数据字节 0~7 的数据字节数应与映射的数据字节数相同 PDO 才会被处理；
- 2) 十进制数 500 对应的 16 进制数为 0x01F4，十进制数 5000 对应的 16 进制数为 0x1388；
- 3) 要使速度单位为 Hz 且分辨率为 0.1Hz，须先确保 0x200A 索引和 0x0077 寄存器（0x4077 索引）的值均为 0。

可通过如下 CAN 消息实现设定输出占空比为 50.0%。

CAN 标识符	数据字节 0~7（16 进制，低字节在前）
0x201	00 F4 01 00 00 00 00 00

可通过如下 CAN 消息实现设定输出闭环调速的换向频率为 500.0Hz。

CAN 标识符	数据字节 0~7（16 进制，低字节在前）
0x201	01 88 13 00 00 00 00 00

可通过如下 CAN 消息实现电机紧急制动。

CAN 标识符	数据字节 0~7（16 进制，低字节在前）
0x201	11 00 00 00 00 00 00 00

示例二：通过 TPDO 映射电机实时状态相关索引，实现 TPDO1 以 500ms 的周期定时传送电机换向频率和电机转动位置数据。

TPDO1 的通讯参数我们先使用默认参数，那么 TPDO1 的 COB-ID 为 0x180+节点 ID；而 TPDO1 的映射参数索引号为 0x1A00，我们先通过对象字典索引号 0x1A00 的子索引号 0 写入单字节数据 0 使映射失效，然后对 1~8 子索引进行如下设置：

索引号	子索引号	数据	映射内容
0x1A00	1	0x21020010	索引号：0x2102 子索引号：0 数据位数：16 对象作用：读取电机换向频率
	2	0x21050020	索引号：0x2105 子索引号：0 数据位数：32 对象作用：读取电机转动位置计数值

设置好映射项后，我们对索引号 0x1A00 的子索引号 0 写入映射项数量 2 使映射生效。为了使 TPDO1 以 500ms 的周期发送，我们还需要通过 TPDO1 通讯参数对象字典索引号 0x1800 的子索引号 5 的写入事件时间值 0x01F4，那么当驱动器节点状态为 Operational 时，将以 500ms 的周期发送 TPDO1 消息。若需要掉电保存该映射参数，我

们可通过对 0x1010 索引的子索引号 2 写入数值 0x65766173 将相关参数储存到驱动器

中。若驱动器的节点 ID 为 1，那么，驱动器发送的 TPDO1 的 CAN 消息格式如下：

CAN 标识符	数据字节 0~1	数据字节 2~5
0x181	电机换向频率（低字节在前）	电机转动位置（低字节在前）

同步（SYNC）

在网络范围内同步（尤其在驱动应用中）：在整个网络范围内当前输入值准同时保存，随后传送（如果需要），根据前一个 SYNC 后接收到的报文更新输出值。

< 主从模式：SYNC 主节点定时发送 SYNC 对象，SYNC 从节点收到后同步执行任务。

< 在 SYNC 报文传送后，在给定的时间窗口内传送一个同步 PDO。

SYNC 报文消息格式如下：

CAN 标识符	数据字节
0x80（默认）	无

CAN 标识符：默认值为 0x80，可通过对象字典 0x1005 索引进行配置。

数据字节：SYNC 报文无数据字节，数据长度为 0。

注：PDO 通讯参数传输类型数值应在 1~240 之间周期性同步才有效。对象字典 0x1006 索引配置的通讯间隔时间应与 SYNC 主节点发送 SYNC 报文的间隔时间一致，若发送 SYNC 报文间隔时间大于 0x1006 索引配置的通讯间隔时间的 1.5 倍，将产生通讯错误应急报文。

应急指示对象（EMCY）

在 CANopen 模式下，当驱动器出现错误时，将以最高优先急发送应急报文，应急报文消息格式如下：

CAN 标识符	数据字节 0-1	数据字节 2	数据字节 3-7
0x80+节点 ID	应急错误代码	错误寄存器	制造商特定的错误区域

CAN 标识符第 0~6 位：驱动器自身的节点 ID（0x01~0x7F）

数据字节 0-1：应急错误代码

数据字节 2：错误寄存器，同对象字典 0x1001 索引中数据

数据字节 3-7：制造商特定的错误区域。其中，字节 4-5 为驱动器相应故障的故障值

应急错误代码（16 进制）功能描述如下表所示。

应急错误代码	代码功能描述
00xx	故障复位或无错误（Error Reset or No Error）
10xx	一般性错误（Generic Error）
20xx	电流（Current）
21xx	电流，设备输入侧（Current, device input side）
22xx	电流，设备内部（Current, inside the device）
23xx	电流，设备输出侧（Current, device output side）
30xx	电压（Voltage）
31xx	主电源电压（Mains voltage）
32xx	设备内部电压（Voltage inside the device）
33xx	输出电压（Output voltage）
40xx	温度（Temperature）
41xx	环境温度（Ambient temperature）

42xx	设备温度 (Device tempearture)
50xx	设备硬件 (Device hardware)
60xx	设备软件 (Device software)
61xx	内部软件 (Internal software)
62xx	用户软件 (User software)
63xx	数据集 (Data set)
70xx	附加模块 (Additional modules)
80xx	监视 (Monitoring)
81xx	通讯 (communication)
8110	CAN过载 (CAN overrun)
8120	被动错误 (Error Passive)
8130	节点守卫错误或心跳错误 (Life Guard Error or Heartbeat Error)
8140	从离线恢复 (Recovered from Bus-Off)
82xx	协议错误 (Protocol Error)
8210	由于长度错误PDO未处理 (PDO no processed Due to length error)
8220	超出长度 (Length exceedd)
90xx	外部错误 (External error)
F0xx	附加功能 (Additional functions)
FFxx	设备特定 (Device specific)

其中，驱动器致命错误的应急错误代码（16 进制）定义如下表。

应急错误代码	代码功能描述	故障值
81xx	通讯故障	
2210	制动电流异常	母线电流与电机相电流差值
2310	输出过流	过流关断触发电流值
2320	输出短路/过流	短路/过流关断触发电流值
3110	电源过压	过压触发电压值
3120	电源欠压	欠压触发电压值
4210	驱动器过热	过热触发温度值

当驱动器出现致命错误由Operational状态转换为Pre_Operational状态后，可通过NMT控制模块服务（见 7.5.2 小节）重启通讯清除错误并转换为Operational状态。

基本对象字典（OD）

驱动器支持的 CANopen 基本对象字典（Object Dictionary）索引如下：

索引号	子索引	对象作用	类型	默认值	权限	描述
0x1000	0	设备类型	U32	0x0L	RO	

0x1001	0	错误寄存器	U8	0x00	RO	Bit M/O Meaning 0 M 一般性错误 1 O 电流 2 O 电压 3 O 温度 4 O 通行错误 5 O 设备子协议 6 O 保留 7 O 制造商特定错误
0x1002	0	制造商状态寄存器	U32	0x0L	RO	
0x1003	错误集					记录设备出现的错误并通过 ENCY 发送出去的错误
	0	错误数量	U8	0x0L	RW	出现错误的总数，写 0 时清除历史数据
	1	最近一次出现的 错误记录	U32	0x0L	RO	
	2	最近第二次出现的 错误记录	U32	0x0L	RO	
		U32	0x0L	RO	
	8	最近第八次出现的 错误记录	U32	0x0L	RO	
0x1005	0	SYNC 的COB-ID	U32	0x80L	RW	Bit30: 1 设备产生SYNC, 0 设备 不产生SYNC Bit29: 1 29 位ID, 0 11 位ID Bit29-0: 29 位ID 或 11 位ID (低 11 位)
0x1006	0	SYNC 通信间隔时间	U32	0x0L	RW	该 32 位数据为间隔时间，以 uS 为单位。
0x1007	0	同步窗口长度	U32	0x0L	RW	PDO 同步时间窗口，以 uS 为单位。
0x1008	0	制造商设备名称	VSTR		RO	16 字节字符串
0x1009	0	制造商硬件版本	VSTR	“0.00”	RO	无意义
0x100A	0	制造商软件版本	VSTR		RO	4 字节字符串
0x1010	存储参数					储存参数时可写入“save”到对应的字典中，s 放在低 8 位。
	0	最大子索引号	U8	2	RO	
	1	存储所有参数	U32	0x1L	RW	写入“save” (0x65766173) 储存所有参数 (1000h-9FFFh)
	2	储存通信参数	U32	0x1L	RW	写入“save” (0x65766173) 储存CANopen 参数 (1000h-2FFFh)
0x1011	恢复默认参数					恢复默认参数可写入“load”到对应的字典中，l 放在低 8 位。设备在复位或重新上电时才能恢复参数。
	0	最大子索引号	U8	0x02	RO	
	1	恢复所有参数	U32	0x1L	RW	写入“load” (0x64616f6c) 恢复所有参数 (1000h-9FFFh)
	2	恢复通信参数	U32	0x1L	RW	写入“load” (0x64616f6c) 恢复CANopen 参数 (1000h-2FFFh)

0x1014	0	EMCY 报 文 COB-ID	U32	0x80L	RO	
0x1015	0	EMCY 禁止时间	U16	0x00	RW	
0x1016	消费心跳包时间					
	0	最大子索引号	U32	0x0L	RO	
	1	消费心跳包时间	U32	0x0L	RW	消费心跳包时间格式： Bit31-24 保留（00）； Bit23-16 Node-ID， unsigned8； Bit15-0 heartbeat time（ms）， unsigned16
	2	消费心跳包时间	U32	0x0L	RW	
	3	消费心跳包时间	U32	0x0L	RW	
0x1017	0	产生心跳时间 （ms）	U16	0x00	RW	
0x1018	标识对象					
	0	最大子索引号	U8	0x04	RO	
	1	厂家标识	U32	0x0L	RO	
	2	产品代码	U32	0x0L	RO	
	3	修订版本号	U32	0x0L	RO	
	4	序列号	U32	0x0L	RO	
0x1029	0	错误行为	U8	0x00	RO	8 位数据值定义如下： 0， pre-operational（Only if the current state is operational） 1， no state change 2， stopped
0x1200	服务器 SDO 参数					
	0	最大子索引号	U8	0x02	RO	
	1	COB-ID Client->Sever （RX）	U32	0x600L	RO	
	2	COB-ID Sever->Client （TX）	U32	0x580L	RO	
0x1400 0x1403	RPDOx (x=1,2,3,4)通讯参数					
	0	最大子索引号	U8	0x02	RO	
	1	RPDOx 使用的 COB-ID	U32	0x200L 0x300L 0x400L 0x500L	RW	Bit31:1 存在PDO,0 不存在PDO Bit30: 1 no RTR allowed on this PDO, 0 RTR allowed on this PDO Bit29: 1 29 位 ID, 0 11 位 ID Bit29-0: 29 位 ID 或 11 位 ID (低 11 位)
	2	传输类型	U8	0xFE	RW	
0x1600 0x1603	RPDOx (x=1,2,3,4)映射参数					
	0	最大子索引号	U8	0x00	RW	范围 0~8
	1	RPDOx 映射	U32	0x0L	RW	Bit31-16: 索引号 Bit15-8: 子索引号 Bit7-0: 对象字节数
	2	RPDOx 映射	U32	0x0L	RW	
	3	RPDOx 映射	U32	0x0L	RW	
	4	RPDOx 映射	U32	0x0L	RW	
	5	RPDOx 映射	U32	0x0L	RW	

	6	RPDOx 映射	U32	0x0L	RW	
	7	RPDOx 映射	U32	0x0L	RW	
	8	RPDOx 映射	U32	0x0L	RW	
0x1800 0x1803	TPDOx (x=1,2,3,4)通讯参数					
	0	最大子索引号	U8	0x05	RO	
	1	TPDOx 使用 的 COB-ID	U32	0x180L 0x280L 0x380L 0x480L	RW	Bit31: 1 存在 PDO, 0 不存在 PDO Bit30: 1 no RTR allowed on this PDO, 0 RTR allowed on this PDO Bit29: 1 29 位 ID, 0 11 位 ID Bit29-0: 29 位 ID 或 11 位 ID (低 11 位)
	2	传输类型	U8	0xFE	RW	
	3	禁止时间 (100us)	U16	0x00	RW	
	4	兼容入口 (保留)	U8	0x00	RW	
	5	事件时间 (ms)	U16	0x00	RW	
	TPDOx (x=1,2,3,4)映射参数					
0x1A00 0x1A03	0	最大子索引号	U8	0x00	RW	范围 0~8
	1	TPDOx 映射	U32	0x0L	RW	Bit31-16: 索引号 Bit15-8: 子索引号 Bit7-0: 对象字节数
	2	TPDOx 映射	U32	0x0L	RW	
	3	TPDOx 映射	U32	0x0L	RW	
	4	TPDOx 映射	U32	0x0L	RW	
	5	TPDOx 映射	U32	0x0L	RW	
	6	TPDOx 映射	U32	0x0L	RW	
	7	TPDOx 映射	U32	0x0L	RW	
0x1F80	0	NMTStartup	U32	0x0L	RW	

注：CANopen 基本对象字典适用于 CANopen 通讯模式

对象字典定义

电机控制对象字典

索引号	子索引	对象作用	类型	值	权限	描述
0x2000	0	电机控制类型	U8	0x00~0x03, 0x10~0x12	RW/E	0x00: 占空比调速 0x01: 闭环调速 0x02: 力矩控制 3: 位置闭环 0x10: 正常停止 0x11: 紧急停止 0x12: 自由停止

0x2001	0	电机控制量	S16 或 S32	占空比： -1000~1000 速度/位置闭环： -最大速度~最大速度 力矩控制： -最大电流~最大电流	RW	占空比调速方式时，写入数值乘以 0.1%为输出占空比； 闭环调速或位置控制时，当 0x200A 索引对象数值为 1 时，写入数值为电机目标转速(RPM)；当 0x200A 索引对象数值为 0 时，写入数值为电机目标换向频率(Hz)，若 0x0077 寄存器值为 0，则设写入数值乘以 0.1 为目标换向频率； 力矩控制方式时，写入数值乘以 0.01 为目标电流 (A) 。
0x2002	0	位置类型	U8	0,1	RW	0: 绝对位置 1: 相对位置
0x2003	0	目标位置	S32		RW	
0x2004	0	开环调速 PWM 上升缓冲时间	U8	0~255	RW	数值乘以 0.1 为输出占比空由 0 增加到 100.0%所需时间
0x2005	0	开环调速 PWM 下降缓冲时间	U8	0~255	RW	数值乘以 0.1 为输出占比空由 100.0%减小到 0 所需时间
0x2006	0	闭环调速加速加速度	U16	1~65535	RW	数值乘以 0.1 为换向频率增大速度，单位为 Hz/s
0x2007	0	闭环调速减速加速度	U16	1~65535	RW	数值乘以 0.1 为换向频率减小速度，单位为 Hz/s
0x2008	0	最大速度	U16	1~65535	RW	当 0x200A 索引对象数值为 1 时，写入数值为电机最大转速(RPM)；当 0x200A 索引对象数值为 0 时，写入数值为电机最大换向频率 (Hz)，若 0x0077 寄存器值为 0，则设写入数值乘以 0.1 为换向频率；
0x200A	0	速度单位	U8	0,1	RW/E	0: Hz 1: RPM 若要设置速度单位为 RPM，须先通过 0x0073 寄存器配置电机极个数
0x200B	0	加速度单位	U8	0,1	RW/E	0: Hz/s 1: Rad/s ² 若要设置加速度单位为 Rad/s ² ，须先通过 0x0073 寄存器配置电机极个数

0x200F	0	重新设定电机转动位置计数值	S32		WO	若写0 则对当前计数值清零
--------	---	---------------	-----	--	----	---------------

注：/E 表示可通过 0x1010 索引保存到内部存储器中。

实时状态对象字典

索引号	子索引	对象作用	类型	值	权限	备注
0x2100	0	电机相电流	U16		RO	数值乘以 0.01 为电流值，单位为A
0x2101	0	输出 PWM	S16	-1000~1000	RO	数值乘以 0.1%为占空比
0x2102	0	电机换向频率	S32		RO	数值乘以 0.1 为换向频率
0x2105	0	电机转动位置计 数值	S32		RO	
0x2106	0	电机位置控制完 成状态	U8	0,1	RO	0: 未完成 1: 完成
0x2109	0	相对转动位置计 数值	S32		RO	
0x210A	0	电机转速	U32		RO	单位 RPM。
0x210B	0	完成位置控制剩 余时间	U32		RO	单位为ms
0x210C	0	母线电流	U16		RO	数值乘以 0.01 为母线电流，单位为A
0x210D	0	电源电压	U16		RO	数值乘以 0.1 为电源电压，单位为V
0x210E	0	IN1~IN3 电平	U8		RO	每一位代表一路输入的电平
0x021F	0	功率管附近温度	U16		RO	数值乘以 0.1 为温度，单位为℃
0x2111	0	电机堵转状态	U8	0,1,2	RO	0: 未堵转 1: 正转堵转停止 2: 反转堵转停止
0x2112	0	故障状态	U8	0~10	RO	0: 无错误 1: 尚未学习 2: 堵转停止 3: 霍尔错误 4: 无法达到目标速度 5: 保留 6: 过流关断 7: 过热关断 8: 过压关断 9: 欠压关断 10: 短路/过流关断
0x2113	0	VO 输出状态	U8	0,1	RO	0: 关断 1: 打开 可通过 0x7000 寄存器设置
0x2114	0	SQ1、SQ2 限位接 口电平	U8	0x00~0x03	RO	第0,1 位分别为SQ1 和 SQ2 电平

0x2115	0	霍尔信号电平	U8	0x00~0x07	RO	第 0、1、2 位分别为 HU、HV、HW 的电平
0x2116	0	拨码开关状态	U8	0x00~0xFF	RO	第0~7 位依次为拨码开关的第 1~8 位的状态, 0 为 OFF, 1 为ON
0x2120	0	IN1 输入类型	U8	0,1	RO	0: 模拟量 1:数字量 可通过 0x7002 寄存器设置
0x2121	读取 INx 脉冲信号输入值					
	0	最大子索引号	U8	3	RO	
	1	IN1 输入 PWM	U16		RO	数值乘以 0.1%为占空比
	2	IN1 输入频率	U16		RO	单位 Hz
	3	IN1 输入脉冲个数	S32		RO	可通过 0x7004 寄存器写 1 清零
0x2122	读取 INx 模拟信号输入值					
	0	最大子索引号	U8	3	RO	
	1	IN1 电压	U16		RO	单位 mV
	2	IN2 电压	U16		RO	单位 mV
	3	IN3 电压	U16		RO	单位 mV

通讯参数对象字典

索引号	子索引	对象作用	类型	值	权限	描述
0x2201	0	CAN 节点 ID	U8	1~127	RW/E	修改后在复位通讯后生效
0x2202	0	CAN 波特率	U8	0~7	RW/E	0: 10kbps 1: 20kbps 2: 50kbps 3: 125kbps 4: 250kbps 5: 500kbps 6: 800kbps 7: 1Mbps 修改后在复位通讯后生效
0x2203	0	SYNC 计数	U16		RW	
0x2004	0	SYNC 时间	U16		RO	

注：/E 表示可通过 0x1010 索引保存到内部存储器中。

常见问题和注意事项

1. 常见问题

- 1) 开关(包括限位开关)或按钮接线较长时，并没有操作开关或按钮，驱动器出现误动作，操作开关或按钮响应不灵。

答：这可能是由开关或按钮信号线上的干扰引起的，建议在各信号线上加上几K的上拉电阻到VO，或使用屏蔽电缆。

- 2) 485 通讯方式下主站无法与驱动器通讯。

答：请检查主站串口波特率、校验方式、从站地址是否与驱动器配置的一致，485 通讯接线是否正确，485 主站与从站间应是按 A-A、B-B 方式连接的，检测帧格式是否正确。如果主站是 PC 机，可以先使用 modbus 调试工具测试通讯是否正常。

- 3) 电机堵转时，电机一直震动，启用了堵转停转功能并不会停转。

答：可将额定电流参数配置大些；如果使用 485 配置参数，还可配置工作电流为之前的额定电流值。

- 4) 电机在某些转动角度能够直接启动，而在某些角度不能直接启动，需要外力稍转动一下才能启动。

答：请检查电机电源线是否均连接牢固，驱动器输出保险丝是否有损坏（须将保险丝拆下来测量，不可直接测量保险丝座）。

2. 故障报警处理

当电机控制过程中出现异常后，驱动器将发出报警声，并可通过 0x0033 寄存器读取相关的故障代码，故障说明如表 8.1 所示。

表 8.1 故障相关说明

蜂鸣器连续鸣叫次数	故障代码	故障描述	处置办法
1	0x01	未学习电机相序	电机空载条件下对电机进行相序学习，若无法学习成功，请参照故障代码 0x03 的处置办法处理后再学习相序。

2	0x03	霍尔信号故障	<p>1) 若电机无法转动并报警，请检查电机霍尔信号接线是否正确、牢固；通过外力转动电机时，检查电机每根霍尔输出信号电压幅值变化是否正常；</p> <p>2) 若电机转动过程中偶尔报警，请检查电机霍尔信号接线是否牢固；将电机霍尔信号电缆与电机线圈电源电缆分离远一些走线；若电机转速在 10000RPM 以下，可在每一霍尔信号线并联一个 103 的电容到霍尔信号地；若无需位置控制，可将霍尔错误屏蔽时间寄存器（0x0105）值配置大一些。</p>
3	0x04	达不到目标速度	<p>1) 检查给定驱动器的最大换向频率是否超过了电机在满 PWM 情况下能达到的最大换向频率；</p> <p>2) 检查电机负载是否过大，驱动器输出的相电流是否达到了配置的最大负载电流；</p> <p>3) 是否最大负载电流配置过小，加速度配置过大。</p>
4	0x02	堵转停止	检查电机负载是否过大，电机被堵转。
5	0x05	保留	
6	0x06	过流关断	<p>1) 检查是否配置的电机额定电流太小，而电机标示的额定电流或额定功率却较大；</p> <p>2) 检查配置的过流关断值是否太小，导致电机正常工作情况下也触发该故障报警；</p> <p>3) 检查电机电源线是否接触良好或存在搭线短路；</p> <p>4) 是否电机在高速转动情况下突然被卡住。</p>
7	0x08	过压关断	<p>1) 检查电源空载时的电压是否超出配置的过压关断电压；</p> <p>2) 检查电机由转动切换为制动的过程中，电源电压是否突然升高超过配置的过压关断电压。</p>
8	0x09	欠压关断	<p>1) 检查电机带负载转动过程中，电源电压是否存在跌落低于配置的欠压关断电压情况，电源功率太小或电源稳压响应慢；</p> <p>2) 检查电源线是否过长过细，电机带负载转动过程中电源线上产生较大压降。</p>
9	0x07	过热关断	<p>1) 驱动器是否通风良好；</p> <p>2) 驱动器是否长时间过载或高温环境下工作；</p> <p>3) 配置的过热关断温度是否太低。</p>
10	0x0A	过流/短路关断	首先检查是否存在搭线/短路故障，其次按照故障代码 0x06 处置办法进行检查。

3. 注意事项

- 1) 驱动器电源电压应在 9~60V 之间。若电压超压，上电后可能烧毁驱动器。
- 2) 驱动器与不带隔离的用户控制器(信号线)相连时，电源请勿共地，否则存在安全隐患可能损坏驱动器或用户控制器。
- 3) 由于控制信号线很脆弱，在使用过程中，控制信号的任何信号线都不能与电源或电机接口的接线搭在一起，否则极可能烧掉驱动器，且难以维修。

- 4) 电源或电机接口的接线千万不要与电位器、限位或通讯接口搭在一起，否则可能烧掉驱动器部分器件。电源地或控制信号的地也不要与机壳相连，否则可能造成驱动器工作不稳定。如果有条件，机壳请与大地相连。
- 5) 驱动器掉电的时候，不要直接或间接高速旋转电机，否则电机产生的电动势可能烧掉驱动器。
- 6) 驱动器应先与电机连接好后才上电，否则可能烧掉保险丝或驱动器。
- 7) 电机接口千万不能短路，否则可能烧掉保险丝或驱动器。
- 8) 注意驱动器不要受潮，不要让驱动器板上的元件短路，不要用手触摸板上元件的引脚和焊盘。
- 9) 如果驱动器上的保险丝在使用时烧毁，请检查线路，正确连接。保险丝烧毁后，不可强行接通电源，继续使用；否则驱动器会严重烧毁、无法维修。
- 10) 在驱动器发生故障时，用户应及时与本公司联系，不得私自维修和更换配件。
- 11) 本款驱动器只能用于驱动感性负载（如电机），不能用于驱动阻性（如电阻）或容性负载（如电容）。
- 12) 请用户仔细阅读注意事项及保修说明，这样会为您减少不必要的麻烦。
- 13) 请用户仔细阅读此 ，正确使用本款驱动器。

使用Windows自带的计算器进行十进制 – 十六进制转换

1.使用 Windows XP 自带计算器进行十进制到十六进制的转换步骤如下：

16) 打开系统自带的计算器工具，如 图 10.1 所示。



图 10.1 Windows XP 自带的计算器

17) 选择“查看” – “科学型”菜单项，计算器界面将切换为如 图 10.10.2 所示。



图 10.10.2 计算器界面切换科学型后的界面

18) 点击“十进制”单选按钮，输入要转换为十六进制的十进制数，我们以-100 为例进行说明，通过右边数字按钮首先输入 100，然后按“+/-”按钮输入负号，如 图 10.3 所示。



图 10.3 在计算器里输入“-100”

- 19) 然后点击左边“十六进制”单选按钮，此时，我们先前输入的十进制数-100 被转换为 int64 类型的整数以十六进制进行显示。如果我们要以long型、short型或char型整数的十六进制进行显示，可分别按右边的“双字”、“单字”和“字节”单选按钮进行显示，显示结果如图 10.4 所示。



图 10.4 “-100”转换为 short 型并以十六进制显示

- 2.使用 Windows 7 自带计算器进行十进制到十六进制的转换步骤如下：
- 20) 打开系统自带的计算器工具，如图 10.5 所示。



图 10.5Windows 7 自带的计算器

- 21) 选择“查看”–“程序员”菜单项，计算器界面将切换为如图 10.6 所示。



图 10.6 计算器界面切换科学型后的界面

- 2) 点击“十进制”单选按钮，输入要转换为十六进制的十进制数，我们以-100 为例进行说明，通过右边数字按钮首先输入 100，然后按“±”按钮输入负号，如图 10.7 所示。



图 10.7 在计算器里输入“-100”

- 2) 然后点击左边“十六进制”单选按钮，此时，我们先前输入的十进制数-100 被转换为 int64 类型的整数以十六进制进行显示。如果我们要以long型、short型或char型整数的十六进制进行显示，可分别按左下方的“双字”、“字”和“字节”单选按钮进行显示。如图 10.8 所示。



图 10.8 “-100”转换为 short 型并以十六进制显示

CRC16 的计算

C语言CRC生成函数如 程序 10.1 所示。所有的可能的CRC 值都被预装在两个数组中，当计算报文内容时可以简单的索引即可。一个数组含有 16 位CRC 域的所有 256 个可能的高位字节，另一个数组含有地位字节的值。这种索引访问CRC 的方式提供了比对报文缓冲区的每个新字符都计算新的CRC 更快的方法。

注意: 此函数内部执行高/低 CRC 字节的交换。此函数返回的是已经经过交换的 CRC 值。也就是说，从该函数返回的 CRC 值可以直接放置于报文用于发送。

函数使用两个参数：

unsigned char *puchMsg; 指向含有用于生成 CRC 的二进制数据报文缓冲区的指针。

unsigned short usDataLen; 报文缓冲区的字节数。

注：以下 CRC16 生成函数程序摘自 <MODBUS over Serial Line Specification and Implementation Guide V1.02>。

程序 10.1 CRC16 生成函数程序清单

```
/* 高位字节的 CRC 值*/
static unsigned char auchCRCHI[] = {
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81,
    0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0,
    0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01,
    0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81,
    0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0,
    0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01,
    0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81,
    0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0,
    0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01,
    0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80,
    0x41,
    0x40
};

/* 低位字节的 CRC 值*/
static char auchCRCLo[] = {
    0x00, 0xC0, 0xC1, 0x01, 0xC3, 0x03, 0x02, 0xC2, 0xC6, 0x06, 0x07, 0xC7, 0x05, 0xC5, 0xC4,
    0x04, 0xCC, 0x0C, 0x0D, 0xCD, 0x0F, 0xCF, 0xCE, 0x0E, 0x0A, 0xCA, 0xCB, 0x0B, 0xC9, 0x09,
    0x08, 0xC8, 0xD8, 0x18, 0x19, 0xD9, 0x1B, 0xDB, 0xDA, 0x1A, 0x1E, 0xDE, 0xDF, 0x1F, 0xDD,
    0x1D, 0x1C, 0xDC, 0x14, 0xD4, 0xD5, 0x15, 0xD7, 0x17, 0x16, 0xD6, 0xD2, 0x12, 0x13, 0xD3,
    0x11, 0xD1, 0xD0, 0x10, 0xF0, 0x30, 0x31, 0xF1, 0x33, 0xF3, 0xF2, 0x32, 0x36, 0xF6, 0xF7,
    0x37, 0xF5, 0x35, 0x34, 0xF4, 0x3C, 0xFC, 0xFD, 0x3D, 0xFF, 0x3F, 0x3E, 0xFE, 0xFA, 0x3A,
```

```
0x3B, 0xFB, 0x39, 0xF9, 0xF8, 0x38, 0x28, 0xE8, 0xE9, 0x29, 0xEB, 0x2B, 0x2A, 0xEA, 0xEE,
0x2E, 0x2F, 0xEF, 0x2D, 0xED, 0xEC, 0x2C, 0xE4, 0x24, 0x25, 0xE5, 0x27, 0xE7, 0xE6, 0x26,
0x22, 0xE2, 0xE3, 0x23, 0xE1, 0x21, 0x20, 0xE0, 0xA0, 0x60, 0x61, 0xA1, 0x63, 0xA3, 0xA2,
0x62, 0x66, 0xA6, 0xA7, 0x67, 0xA5, 0x65, 0x64, 0xA4, 0x6C, 0xAC, 0xAD, 0x6D, 0xAF, 0x6F,
0x6E, 0xAE, 0xAA, 0x6A, 0x6B, 0xAB, 0x69, 0xA9, 0xA8, 0x68, 0x78, 0xB8, 0xB9, 0x79, 0xBB,
0x7B, 0x7A, 0xBA, 0xBE, 0x7E, 0x7F, 0xBF, 0x7D, 0xBD, 0xBC, 0x7C, 0xB4, 0x74, 0x75, 0xB5,
0x77, 0xB7, 0xB6, 0x76, 0x72, 0xB2, 0xB3, 0x73, 0xB1, 0x71, 0x70, 0xB0, 0x50, 0x90, 0x91,
0x51, 0x93, 0x53, 0x52, 0x92, 0x96, 0x56, 0x57, 0x97, 0x55, 0x95, 0x94, 0x54, 0x9C, 0x5C,
0x5D, 0x9D, 0x5F, 0x9F, 0x9E, 0x5E, 0x5A, 0x9A, 0x9B, 0x5B, 0x99, 0x59, 0x58, 0x98, 0x88,
0x48, 0x49, 0x89, 0x4B, 0x8B, 0x8A, 0x4A, 0x4E, 0x8E, 0x8F, 0x4F, 0x8D, 0x4D, 0x4C, 0x8C,
0x44, 0x84, 0x85, 0x45, 0x87, 0x47, 0x46, 0x86, 0x82, 0x42, 0x43, 0x83, 0x41, 0x81, 0x80,
0x40
};
unsigned short CRC16(puchMsg, usDataLen)    /* 函数以 unsigned short 类型返回CRC */
unsigned char *puchMsg,                    /* 用于计算 CRC 的报文*/
unsigned short usDataLen                    /* 报文中的字节数*/
{
    unsigned char uchCRCHi = 0xFF;          /* CRC 的高字节初始化*/
    unsigned char uchCRCLo = 0xFF;          /* CRC 的低字节初始化*/
    unsigned uIndex ;                       /* CRC 查询表索引*/

    while (usDataLen--)                     /* 完成整个报文缓冲区*/
    {
        uIndex = uchCRCLo ^ *puchMsg++;     /* 计算 CRC */
        uchCRCLo = uchCRCHi ^ uchCRCHi[uIndex];
        uchCRCHi = uchCRCLo[uIndex];
    }
    return (uchCRCHi << 8 | uchCRCLo);
}
```


免责声明

本文档提供相关产品的使用说明。本文档并未授予任何知识产权的许可，并未以明示或暗示，或以禁止发言或其它方式授予任何知识产权许可。并且，本产品的销售和 / 或使用我们不作任何明示或暗示的担保，包括对产品的特定用途适用性、适销性或对任何专利权、版权或其它知识产权的侵权责任等，均不作担保。我们可能随时对产品规格及产品描述做出修改，恕不另行通知。

成都市装甲力量科技有限公司