1 ID200 系列伺服驱动器串口通讯协议(ModeBus_Rtu 协议)

1.1 通讯接口定义

SDB 驱动器面板上的 CN3、CN4 接口是网络通讯插座（母头），与插座关联的外部通讯电缆接口（公头）示意如下图 1-1、1-2 所示：

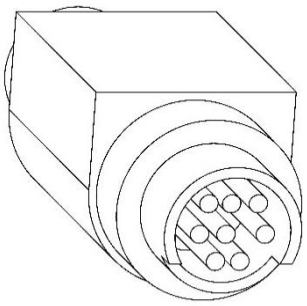

图 1-1 正面外观视图

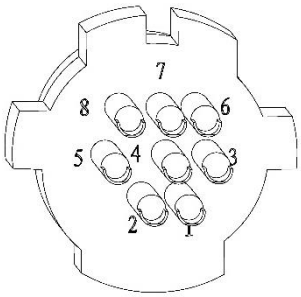

图 1-2 背面焊接视图

表 1-1 CN3、CN4 引脚定义表

信号名称	引脚	符号	功能	备注
数字电源	1	+5V	+5V	—
RS232 接收信号	2	232R	接收数据	仅 CN3-2、CN3-3 有 RS232 信号。CN4-2 脚用于 RS485 终端匹配,CN4-3 脚用于 CAN 总线匹配。
RS232 发送信号	3	232T	发送数据	
RS485 SA 信号	4	485/SA	信号 485+	CN3、CN4 共有互联
数字地	5	GND	公共端	

RS485 SB 信号	6	485/SB	信号 485-	
CANopen 信号	7	CANH	CAN-high 线	
CANopen 信号	8	CANL	CAN-low 线	
通讯端口金属外壳	-	FG2	与屏蔽连接	与 PE、FG 不连通

1.2 通讯总线网络

ID200 系列伺服驱动器，支持 RS485 通信，RS485 通信采用国际标准的 ModBus RTU 通讯协议进行的主从通讯。用户可通过 PC/PLC、控制上位机等实现集中控制（设定伺服驱动器控制命令、运行速度、相关功能码参数的修改，伺服驱动器工作状态及故障信息的监控等），以适应特定的应用要求。

通讯采用 MODEBUS RTU 模式。

1.2.1 RS-485 异步串口

RS-485 网络组网示意图如图 1-3 所示，1 根 RS485 总线上最多可直接带 32 台伺服驱动器，如果配置 485 总线中继装置，从站总数最多可达到 255 台。

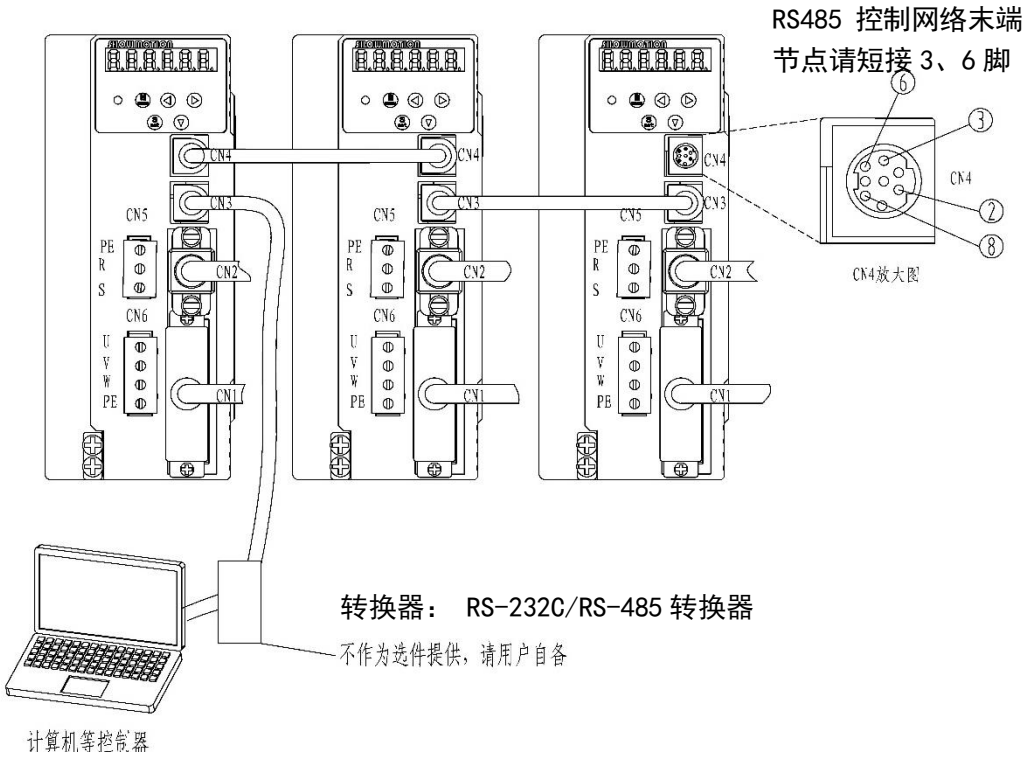

图 1-3 RS485 网络示意图

RS485 网络具体接线图如图 1-4 所示：

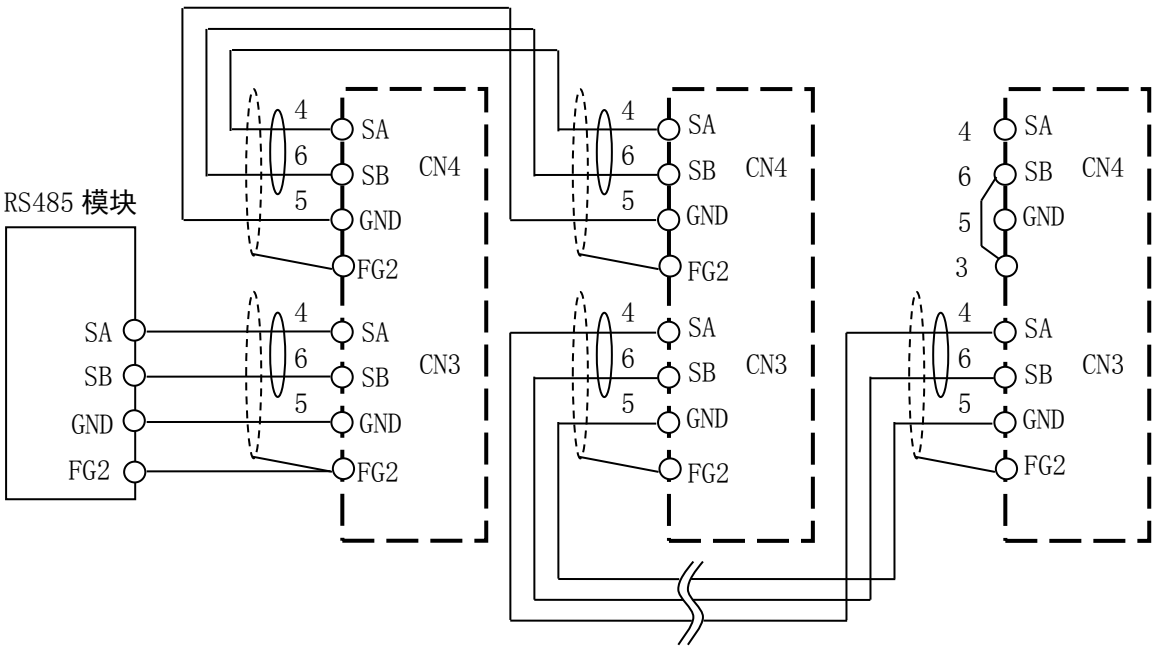

图 1-4 RS485 网络接线图

注意：1、建议使用屏蔽双绞线，屏蔽单点接地，SA、SB 需要双绞。

2、最远端驱动器 CN4 的 3、6 脚请短接（内有匹配电阻）。

3、在一定通讯距离条件下，请选择合适波特率（参数 P00C）。

4、挂接点数>3 个请使用有源 RS485 转换器或中继器。

5、FG2 是指 CN3、CN4 插座的金属外壳。

1.2.2 RS-232C 异步串口

RS-232 网络如图 1-5 所示，使用 RS-232C 通讯功能只能控制 1 台伺服驱动器，仅 CN3 支持 RS-232C，CN4 上无 RS232 信号。

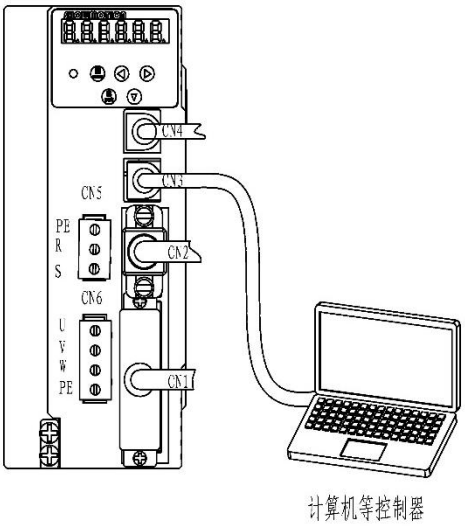

图 1-5 RS232C 网络示意图

RS232C 网络具体接线请按图 1-6 所示进行接线。

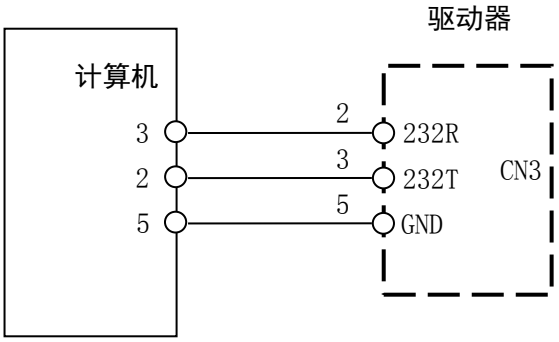

图 1-6 RS232C 网络接线图

注意：1、在一定通讯距离条件下，请选择合适波特率（参数 P00C），最长距离<15m。
2、传统 PC 机串口有 D-SUB9、D-SUB25 两种类型。与 PC 机 D-SUB9 针标准串口连接时通讯线（2-3、3-2）交叉连接，与 PC 机 D-SUB25 针标准串口连接时通讯线（2-2、3-3）连接。

1.3 通信协议规定

1、通讯方式：

RS232C 和 RS485 方式不能同时使用，必须使用通讯参数（P00D）选择使用 RS232C 方式或 RS485 方式，设置如下。

0：RS485 方式；

1：RS232C 方式。

2、帧结构：

1 位起始位，8 位数据位，校验位或无校验，1~2 位停止位，帧结构由参数 P00A 设置。

如图 1-7 所示：

0：1 位停止位、无校验、8 位数据；

1：1 位停止位、偶校验、8 位数据；

2：1 位停止位、奇校验、8 位数据；

3：2 位停止位、无校验、8 位数据。

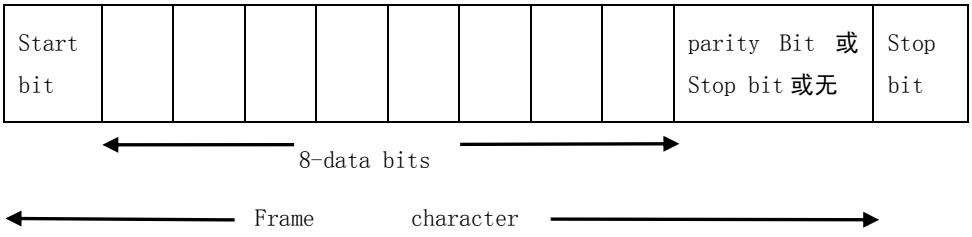

图 1-7 帧结构定义

3、波特率：

通讯的速率由波特率参数（P00C）来决定，具体如下：

0：300bps **1**：600bps **2**：1200bps **3**：2400bps **4**：4800bps

5：9600bps **6**：19200bps **7**：38400bps **8**：57600bps **9**：115200bps

4、帧间隔时间：

ModeBus 通讯时，每次数据的输入输出是由 N 个帧组成的数据包，N 个帧之间最大延时不能超过帧间隔时间。该间隔时间由 P009 参数设定，当两个帧之间延时间隔超过限制时，通讯主站或从站认为该包无效，重启一次新的数据包传输。

帧间隔时间=P009*50ms+10ms。

5、站号：

每个伺服驱动器仅能设定唯一通讯地址。若重复设定通讯地址将导致无法正常通讯。驱动器的站号通过伺服驱动器的站号（参数 P000）来设置。

驱动器编号范围：**1～254**。

编号 0 是广播站号，对所有从站伺服驱动器都有效，从站无需应答。

6、通讯流程：

每次 ModeBus_Rtu 通讯都是由主站发起的，从站根据接收到的数据包内容校验通过后进行应答。当接收的数据包站号为 0x00 时，表示该包对所有从站有效，从站无需应答。

基本流程如图 1-8 所示。

站号	功能码	长度或数据	0~N 数据	校验码

主站发起

站号	功能码	数据长度或数据	0~N 数据	校验码

从站应答

（广播除外）

图 1-8 基本通讯流程

7、数据包定义：

无论是上位机还是驱动器，一次完整的 ModeBus_Rtu 通信单位都是数据包，数据包定义如表 1-2 所示。

表 1-2 数据包的定义

STX	超过 10ms 的静止时段
ADR	站号 ：1-byte
CMD	功能码：1-byte
DATA（N-1）	数据内容：N 个 word=2N 个 byte。N 根据功能码及数据长度而不同。 数据内容可以包含目标地址、数据长度、具体数据中的两种或三种。
・・・・・・	
DATA（0）	
CRC	校验码：2-byte
End 1	超过 10ms 的静止时段

ADR 站号：驱动器的编号范围 0～254，但有一特殊编号 255（十六进制 0x00）表示数据包对所有伺服驱动器都有效，从站无需应答。

CMD 功能码：决定数据包的工作含义，不同的功能码包含不同的数据字节长度。

常见功能码定义：

03H：读取从机保持寄存器数据；

04H：读取从机只读寄存器数据；

06H：给从机写入单字数据；

10H：给从机写入多字数据

DATA 数据：由不同的功能码类型来决定它的长度。

校验码：校验范围包含站号、功能码、0～N 数据，涵盖所有数据内容，CRC16 位校验方式。

1.4 通信协议详解

1.4.1 功能码 03H：读取从站保持寄存器数据

读取指定从站某个地址开始的数据内容，最多 127 个字。

例如：从驱动器 01H 上寄存器地址 0200H 开始读取 2 个 Word 的数据内容：

命令信息	
ADR	01H
CMD	03H
起始数据地址高	02H（高字节）

回应信息	
ADR	01H
CMD	03H
数据字节数	04H

起始数据地址低	00H（低字节）
数据字数	00H
	02H
CRC Low	C5H（低字节）
CRC High	B3H（高字节）

起始数据地址0200H的内容	00H（高字节）
	B1H（低字节）
第二个数据地址0201H的内容	1FH（高字节）
	40H（低字节）
CRC Low	A3H（低字节）
CRC High	D4H（高字节）

注意：1、读取返回字节数是读取字数的 2 倍；

2、CRC16 位校验码是低字节在前，高字节在后。

1.4.2 功能码 04H：读取从站只读寄存器数据

读取指定从站某个地址开始的数据内容，最多 127 个字。

除功能码以外，数据格式同 **1.4.1 命令码 03H：读取从站保持寄存器数据。**

1.4.3 功能码 06H：写单字数据到从站寄存器

向指定从站某个寄存器地址写入数据。

例如：向 01H 驱动器寄存器地址 0200H 写入数据 64H：

命令信息	
ADR	01H
CMD	06H
起始数据地址	02H（高字节）
	00H（低字节）
数据内容	00H（高字节）
	64H（低字节）
CRC Low	89H（低字节）
CRC High	99H（高字节）

回应信息	
ADR	01H
CMD	06H
起始数据地址	02H（高字节）
	00H（低字节）
数据内容	00H（高字节）
	64H（低字节）
CRC Low	89H（低字节）
CRC High	99H（高字节）

注意：1、CRC16 位校验码是低字节在前，高字节在后。

1.4.4 功能码 10H：写多个字数据到从站寄存器指定的寄存器

向某个从站指定寄存器地址处写入多个字的数据内容

例如：向 01H 驱动器寄存器地址 0200H 连续写入两个字的数据 0064H 和 0078H：

命令信息	
ADR	01H
CMD	10H
起始数据地址	02H（高字节）
	00H（低字节）
数据 word 个数	00H（高字节）
	02H（低字节）
数据字节数	04H
数据内容 1	00H（高字节）
	64H（低字节）
数据内容 2	00H（高字节）
	78H（低字节）
CRC Low	ABH
CRC High	32H

回应信息	
ADR	01H
CMD	10H
起始数据地址	02H（高字节）
	00H（低字节）
数据 word 个数	00H（高字节）
	02H（低字节）
CRC Low	40H
CRC High	72H

注意：1、CRC16 位校验码是低字节在前，高字节在后。

附：1

0x0000~0x02FF:系统参数（不可见）

0x0300~0x03FF:伺服电机内部参数

0x0400~0x05FF:伺服控制及试运行参数

0x0600~0x0AFF:伺服驱动器基本参数

0x0B00~0x0BFF:状态监测参数

0x1000~0x1800:数据示波器

0x2000~　　:系统内部参数群（用户不可见）

1.1 伺服电机内部参数表说明：0x0300~0x03FF

伺服电机内部参数地址定义：电机配置参数的数据结构见下表，除了编码器精度是长整型以外，其它均是 int 整型数据。电机属性由工厂下载设置，客户只能读取不能修改。

0x0300:编码器类型　0：多线式 1：省线式；2：绝对式

0x0301:电机极对数

0x0302:编码器精度低 16 位

0x0303:编码器精度高 16 位

0x0304:马达最高速度

0x0305:额定电流 单位 0.01A

0x0306:最大电流 单位 0.01A

0x0307:电流环比例

0x0308:电流环积分

0x0309:电场偏移量

0x030A:电场向量值 UVW

0x030B:保留参数

0x030C:保留参数

0x030D:保留参数

0x030E:保留参数

0x030F:保留参数

0x0310:保留参数

0x0311:电机标识符 电机型号显示，共三屏，每屏 6 个字符，两个字符合成一个 int 字

0x0312:电机标识符

0x0313:电机标识符

0x0314:电机标识符

0x0315:电机标识符

0x0316:电机标识符

0x0317:电机标识符

0x0318:电机标识符

0x0319:电机标识符

1.2 伺服控制及试运行参数表说明：0x0400~0x05FF

本参数表用于伺服试运行或者组网控制。客户向特定寄存器写入参数，完成伺服驱动系

统的试运行及远程控制。

0x0400:复位控制（数据=0x01）

该地址写入 0x01，从站应答后重启复位

0x0401:伺服网络使能控制（等同于内部使能）

该地址写入 0x01 伺服使能，00 使能取消。

0x0402:控制模式切换

模式切换仅在模式 3、4、5 下有效，其它模式下无效。

上电时，驱动器默认外部开关切换有效，一旦使用过网络方式进行模式切换，外部开关模式切换将被禁止，直至模式切换参数为 0、1 以外的数值。

网络方式控制模式切换参数优先于 IO 开关方式。

工作模式	模式切换参数		
	0	1	其它
模式 3 位置/速度：	位置	速度	外部有效
模式 4 速度/转矩：	速度	转矩	外部有效
模式 5 位置/转矩：	位置	转矩	外部有效

0x0403:试运行点动控制

0 停止；1 正转；2 反转；其他值退出点动状态

网络方式控制点动等同于按键操作方式。

0x0404:转动惯量比测试

对该地址写入任意值，表明主站请求从站测量惯量，如果通讯正常并且从站不在使能状态，立即测量系统惯量比。测量的结果需要读取状态参数 0x0B12。

网络方式控制惯量测量等同于按键操作方式。

0x0405:原点复归

对该地址写入任意值，表明主站请求从站进行原点复归动作，如果通讯正常并且从站处于位置模式使能状态，则收到该包后将进行原点复归动作。

网络方式控制原点复归等同于 IO 控制的原点复归。

0x0406:原点设定

对该地址写入任意值，表明主站请求从站进行原点设定动作，如果通讯正常并且从站处于位置模式使能状态，则收到该包后将进行原点设定动作。

网络方式控制原点设定等同于 IO 开关方式。

0x0407:IO 强制输出

执行该命令主站强制从站的 IO 输出为给定的状态，0b00XXXXXX（二进制），如果通讯正常，从站收到该包后将进行 IO 输出控制。

IO 强制输出优先于运行状态的正常输出，覆盖伺服原来的状态输出信号。

0x0408:示波器采样启动

1：停止采样；0：开始采样。

上电开始启动采样，直至发送停止采样命令，这时可以读取采样数据。读取完成后可以再次启动采样。

0x0409:采样数据长度

一次连续采样的点数，最大 512 点。

0x040A:采样通道 1 采样信号选择 00~2F（对应状态信息 0x0B00~0x0B2F）

具体参阅 5.4。

0x040B:采样通道 2 采样信号选择 00~2F（对应状态信息 0x0B00~0x0B2F）

0x040C:采样通道 3 采样信号选择 00~2F（对应状态信息 0x0B00~0x0B2F）

0x040D:采样通道 4 采样信号选择 00~2F（对应状态信息 0x0B00~0x0B2F）

0x040E:采样间隔点数（单位 0.1ms）

设置相邻采样点的时间间隔，缺省值 10（1ms）

0x040F:0x01 单次采样；0x00 连续采样

驱动器内部有 4*512 个字的采样缓冲区，四通道同步采集。单次采样是指采样点数达到采样数据长度后自动停止采样，更新数据需要再次启动开始采样（**0x0408**），可根据需要选择单次采样还是连续循环采样。

0x0410:内部位置段启动

内部位置模式下（P04F=2），网络控制内部可编程位置段的运行、暂停、停止。

0：停止从站当前正在运行的所有段；

1~160：主站启动从站指定的某个有效段，该段的延时参数影响下一步的启动。

0x00ff:主站暂停现有位置段的执行，剩余部分自动保存。

0xff00:主站启动暂停位置段剩余部分的执行。

0xffff：主站启动从站的当前段，当前段执行完毕后，当前段的延时参数影响下一步的启动，详见说明书内部位置控制部分的描述。

查询内部位置段状态需要使用状态读取功能来实现。

注意：内部位置段运行时需要了解运行状态，可以通过读取状态参量（0xB1B:内部位置状态量）来获取电机实际运行状态。

内部位置段运行状态信息包含 2 个字节，字节定义如下。

字节 1：表示驱动器当前正在执行或已经完成的最新段号。1~160：表示正在执行或已经完成的段号。00 表示上电复位后或原点操作后尚未收到位置段启动命令。

字节 2：位置状态 ，由 8 位码表示，定义如下。

位号	名称	逻辑定义	
Bit7	保留	X	
Bit6			
Bit5			
Bit4			
Bit3	RUNS	0:驱动器处于未使能状态	1：驱动器处于使能状态
Bit2	ALMS	0：驱动器当前未报警	1：驱动器当前报警
Bit1	ZPS	0:原点操作未完成或未进行原点操作或原点丢失	1：原点操作完成且工作正常
Bit0	IPS	0：当前位置段未完成	1：当前位置段已完成

0x0411:参数保存

参数保存编号是 1 个整型参数，占用 2 个字节。取值范围 0x600~0xAFF，0xffff。当参数保存命令为 0xffff 时，保存当前参数表所有的参数至 E2PROM。

0x600~0xAFF 仅保存指定编号的一个参数至 E2PROM。

0x0412:恢复出厂参数

该地址写入任意值，从站伺服驱动器参数全部恢复出厂设置。客户请定期做好参数备份。

0x0413:设置内部位置起始段、停止段

高 8 位表示起始段，低 8 位表示结束段。起始段、结束段取值范围： 1~160。

0x0414~：报警清除

驱动器报警时，该地址写入任意值将清除报警。清除后如果故障仍然存在将再次报警。

1.3 伺服驱动器基本参数表说明：（对应伺服说明书 P000~P4ff）

0x600 轴地址

0x601 初始状态

0x602 控制模式选择

0x603 转矩限制选择

0x604 保留

0x605 内部/外部速度选择或转矩选择

0x606 保留

0x607 模拟量通道 1 输出选择

0x608 模拟量通道 2 输出选择

0x609 保留

0x60A 保留

0x60B 编码器类型

0x60C 波特率

0x60D 通讯方式选择

0x60E 保留

0x60F 保留

0x610 位置环第 1 比例增益

0x611 速度环第 1 比例增益

0x612 速度环第 1 积分增益

0x613 保留

0x614 第 1 转矩滤波器时间常数

0x615 速度前馈增益

0x616 速度前馈滤波器时间常数

0x617 伺服使能设置

0x618 位置环第 2 比例增益

0x619 速度环第 2 积分增益

0x61A 速度环第 2 积分增益

0x61B~0x61E 保留

0x61f 自动参数调整

0x620 转动惯量比

0x621~0x635 保留

0x636 参数切换模式

0x637~0x63B 保留

0x63A 模拟输入滤波时间选择

0x63B 模拟量输出通道 1 偏置

0x63C 模拟量输出通道 2 偏置

0x63DJOG 速度设置

0x63E 保留

0x63f 位置环微分

0x640 保留

0x641 位置指令取反

0x642 位置脉冲类型

0x643 保留

0x644 脉冲输出通道选择

0x645 反馈脉冲分频系数

0x646 保留

0x647 保留

0x648 指令脉冲分倍频第 1 分子

0x649~0x64A 保留

0x64B 指令脉冲分倍频分母

0x64C 位置平滑滤波器

0x64D~0x64E 保留

0x64f 内外部脉冲输入选择

0x650 模拟速度指令增益

0x651 保留

0x652 模拟速度指令零漂调整

0x653 第 1 内部速度

0x654 第 2 内部速度

0x655 第 3 内部速度

0x656 第 4 内部速度

0x657 保留

0x658 加速时间设置

0x659 减速时间设置

0x65A S 形加减速时间设置

0x65B 内部转矩指令

0x65C 转矩指令增益

0x65D 保留

0x65E 转矩指令零漂调整

0x65F 第 1 转矩限制

0x660~0x669 保留

0x66A 松闸延迟时间

0x66B 抱闸延迟时间

0x66C~0x66E 保留

0x66F 制动电阻的功率

0x670~0x67C 保留

0x67D 最高速度限制

0x67E 转矩模式下速度限制

0x67F~0x8F 制造商参数

0x690 内部位置起始段号

0x691 内部位置结束段号

0x692 内部位置循环模式选择

0x693 内部位置基准方式

0x694 传动方式

0x695 多轴应用标志

0x696 工作原点偏移量低字

0x697 工作原点偏移量高字

0x698 用户密码

0x699 保留

0x69A CAN 节点号

0x69B CAN 波特率

0x69C~0x69F 保留

0x6A0~0xAFF 内部位置段二维数组[160][7]}

内部位置段二维数组[160][7]每个元素也是整型描述如下：

0x6A0:[0][0]第 1 段整圈数；

0x6A1: [0][1]第 1 段非整圈圈内脉冲数；

0x6A2: [0][2]第 1 段速度

0x6A3: [0][3]第 1 加速度

0x6A4: [0][4]第 1 减速度

0x6A5: [0][5]第 1 段完成后停止时间

0x6A6: [0][6]第 1 段完成后的下一段段号

。。。。。。

0xAF9:[159][0]第 160 段整圈数；

0xAFA:[159][1]第 160 段非整圈圈内脉冲数；

0xAFB:[159][2]第 160 段速度

0xAFC:[159][3]第 160 加速度

0xAFD:[159][4]第 160 加速度

0xAFE:[159][5]第 160 段完成后停止时间

0xAFF:[159][6]第 160 段完成后的下一段段号

1.4 状态监测参数表说明

伺服驱动器运行状态的参量地址见下表。前 40 项状态参量均是 int 整型数据，最后四项是 long 长整型每项占用两个地址。

0xB00:给定力矩电流
0xB01:反馈力矩电流
0xB02:给定励磁电流
0xB03:反馈励磁电流
0xB04:ud电压输出
0xB05:Uq电压输出
0xB06:U相电流输出
0xB07:V相电流输出
0xB08:速度给定
0xB09:速度反馈
0xB0A:脉冲速度
0xB0B:反馈加速度
0xB0C:母线电压
0xB0D:散热器温度
0xB0E:模拟输入1
0xB0F:模拟输入2
0xB10:编码器UVW向量
0xB11:电场角度

0xB12:转动惯量比
0xB13:转动惯量比备用
0xB14:速度环比例
0xB15:速度环积分
0xB16:位置环增益
0xB17:位置环微分
0xB18:速度环积分累计
0xB19:内部位置段号
0xB1A:内部位置速度值
0xB1B:内部位置状态量
0xB1C:增量式转子位置
0xB1D:外部输入IO开关量
0xB1E:外部输出IO开关量
0xB1F:最新报警记录
0xB20:位置跟踪误差
0xB21:绝对式编码器圈数
0xB22:程序版本号
0xB23:固件版本号
0xB24:SVN版本号
0xB25:电流传感器规格
0xB26:力矩滤波系数
0xB27:速度单位
0xB28:绝对值编码低16位
0xB29:绝对值编码高16位
0xB2A:输入脉冲低16位
0xB2B:输入脉冲高16位
0xB2C:反馈脉冲低16位
0xB2D:反馈脉冲高16位
0xB2E:内部给定脉冲低16位
0xB2F:内部给定脉冲高16位}

1.5 数据示波器

0x1000：四通道同步采集，每通道最多 512 点。

受 ModeBus Rtu 协议限制，每次只能读 128 数据，全部数据需要 16 次读取。

1.6 CRC16 校验

ID200 Modbus 数据校验采用 CRC16 查表校验。校验方法和校验数组（分成高低字节）如下：

```
unsigned int aucCRCHi[256] = {
    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, 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,
    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
    };

unsigned int aucCRCLo[256] = {
    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
    };

volatile BIG_U16 usMBCRC16( unsigned int * pucFrame, unsigned int usLen )
{

    unsigned int ucCRCHi = 0xFF;
    unsigned int ucCRCLo = 0xFF;
    unsigned int *pCRCTemp;
    int        temp, iIndex;
    pCRCTemp=pucFrame;
    temp=usLen;
        while( temp-- )
        {
            iIndex = (ucCRCLo ^ ((*( pCRCTemp++ ))&0xff))&0xff;
            ucCRCLo = ( unsigned int )( ucCRCHi ^ aucCRCHi[iIndex] )&0xff;
            ucCRCHi = aucCRCLo[iIndex]&0xff;
        }
     return ( unsigned int )( ucCRCHi << 8 | ucCRCLo );

}
```

//本校验方法采用根据字节内容查表获取，速度最快

//客户也可以采用多项式 0xA001H 循环移位产生,无需校验数组，较为耗时，最终结果一致。