

ST 驱动器的 SCL 使用手册



MOONS'

上海安浦鸣志自动化设备有限公司

目 录

简介.....	5
SCL 是什么?	5
SCL 的详细资料	5
开始.....	7
步骤一：安装软件	7
步骤二：用 ST Configurator 设置您的 ST 驱动器	7
通讯协议	8
上电模式	8
给 EN 输入选择一个定义	8
给 OUT 输出选择一个定义	8
步骤三：熟悉 SCL 指令	8
步骤四：开发应用	9
指令.....	10
缓冲器指令	10
立即指令	10
指令摘要	10
运动指令	11
设置指令	12
缓冲器指令	13
状态指令	13
I/O 指令	14
寄存器指令	14
指令详细说明	15
AC-加速度.....	16
AD-模拟量死区	17
AF-模拟量滤波器	18
AG-模拟量速度增益	19
AI-报警复位输入.....	20
AL-报警代码.....	21
AM-最大加速度.....	22
AO-报警输出	23
AP-模拟量位置增益	24
AR-报警复位（立即）	25
AT-模拟量死区.....	26
AV-模拟量偏移量值	27
AX-模拟量复位（缓冲器）	28
AZ-模拟量零点.....	29
BD-刹车释放延时.....	30
BE-刹车使能延时	31
BO-刹车输出.....	32
BR-波特率.....	33
BS-缓冲器状态	34

CC-改变电流	35
CD-待机电流延时时间	36
CE-通讯错误	37
CF-抗共振滤波器频率	38
CG-抗共振滤波器增益	39
CI-改变待机电流	40
CJ-执行点动	41
CM-控制模式	42
CS-改变点动速度	43
CT-继续	44
DA-定义地址	45
DC 运行到设定距离改变	45
DE-减速度	47
DI-距离/位置	48
DL-定义限位	49
EG-细分	50
FC 运行到设定距离改变	50
FD 运行并检测两个传感器信号	50
FE-编码器跟随	53
FI-输入滤波器	54
FL-运行到指定距离	55
FM-带屏蔽距离的 FS	56
FO 运行期间改变输出状态	56
FP-运动到绝对位置	58
FS-运动到传感器信号有效	59
FY-带安全距离的 FS	60
HG-四次谐波滤波器增益	61
HP-四次谐波滤波器相角	62
HW-跟踪手轮	63
IA-即时模拟量输入	64
IC-即时电流反馈	65
ID-即时距离反馈	66
IF-立即指令格式	67
IH-即时高电平输出	68
IL-即时低电平输出	69
IO-输出状态	70
IP-即时位置反馈	71
IS-即时输入状态反馈	72
IT-即时温度反馈	73
IU-即时电压反馈	74
IV-即时速度反馈	75
JA-点动加速度	76
JC-速度模式第二速度	77
JD-点动禁止	78

JE-点动使能	79
JL-点动减速度	80
JS-点动速度	81
MD-电机禁止	82
ME-电机使能	83
MN-型号编号	84
MO-运动输出	85
PB-上电波特率	86
PC-上电电流	87
PI-上电待机电流	88
PM-上电模式	89
PR-协议	90
PS-暂停	94
RE-重新启动	95
RL-寄存器下载（立即指令）	96
RS-查询状态	97
RU-寄存器上传	98
RV-修正版本	99
SA-保存参数	100
SC-状态代码	101
SF-步进滤波器频率	102
SH-找原点	103
SI-使能输入	104
SJ-停止点动	105
SK-停止并清除缓冲器	106
SO-设置输出	107
SP-设置位置	108
SS-发送串文本	109
ST-停止	110
TD-传输延时	111
VC-改变速度	112
VE-速度	113
WI-等待输入	114
WT-等待时间	115
数字寄存器	116
只读寄存器	116
读/写寄存器	116
数据寄存器分配	116
只读寄存器:a~z	116
读/写寄存器地址: A~Z	117
附录 A: 主机串行通讯	119
主机串行通讯的普通结构	119
硬件	119
COM 口设置	119

通讯协议	120
指令传送（主机→驱动器）:	120
响应传送（驱动器→主机）:	120
协议设置（PR 指令）:	120
通讯详细说明	120
传输延时（TD 指令）:	120
通讯信息包	120
驱动器启动:	120
通讯错误	121
硬件错误:	121
解析错误:	121
附录 B: LED 显示代码	121

简介

首先感谢您购买鸣志驱动产品。我们希望我们的产品在性能、价格以及易于使用等方面能最好的满足于你的应用

串行通讯指令语言 SCL 能够用于大多数 MOONS' 驱动器，这份 SCL 使用手册包括的指令主要针对 ST 系列驱动器。

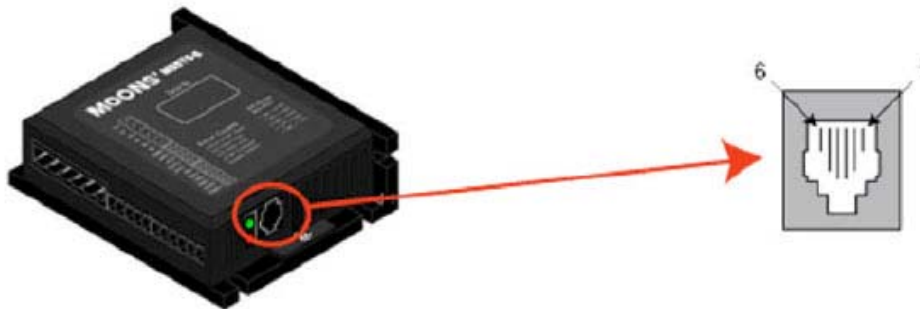
注意：SCL 指令用于其它 MOONS' 驱动器的详细介绍在其它的使用手册上，请访问我们的网站 www.moons.com.cn 查看或下载这些手册。

注意：这份手册只包括 ST 驱动器应用 SCL 指令的详细资料。关于 ST 驱动器其它方面的应用，包括硬件设置、I/O、和软件设置，请查看 ST 用户手册和 ST Configurator 软件的帮助文件。这些手册和软件可以到 www.moons.com.cn 网站下载。

SCL 是什么？

SCL 给用户一个简单的通过串行端口控制电机驱动器的方法。这样就不需要独立的运动控制器或分度器来给驱动器提供脉冲/方向信号，同时也提供了一个简单的面向其它各种工业设备的应用途径，例如：通常配备有标准串行通讯端口的工业设备 PLC 和 HMI。

ST 驱动器配备了一个 RJ-11 插孔的 RS-232 串行端口，如下图所示。



若要在某个应用中使用 SCL，您需要一个主机，例如 PC 机、PLC 或 HMI，与驱动器的串行端口连接，并通过这个连接来发送指令给驱动器。SCL 定义了丰富的指令，包括步进电机的运动指令、驱动器的三个数字输入口、一个模拟量输入口和数字输出口的设置、查询指令。同样也包括其它方面的设置、查询指令，例如电机电流和细分的设置、查询指令。

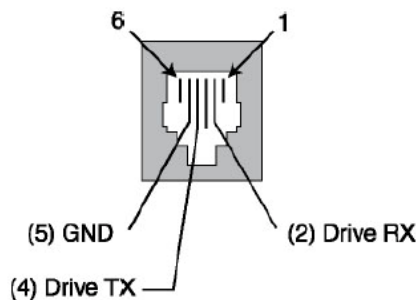
在 SCL 模式下，ST 驱动器从主机接收缓冲器指令，并储存在指令缓冲器，然后立即在缓冲器外执行这些指令，但是，您不能在 ST 驱动器里建立一个可以让驱动器独立运行的驻留程序。如果想这样，您必须选择被设计有程序驻留功能的 MSST-Q 或 MSST-SI 驱动器。

SCL 的详细资料

SCL 的串行通讯由两个基本部分组成：驱动器和主机间的物理连接和主机和驱动器的通讯协议。驱动器和主机之间的物理连接是基于标准的 RS-232 连接，若是 PC 机，可以使用计算机上的九针 COM 口，如果是 PLC 或 HMI，请寻找标有 RS-232、PLC 口、AUX 口、ASCII、或类似的标志 RS-232 接口的连接口。驱动器和主机之间只有三根连接线：发送(TX)、接收(RX)和信号地。

下图是 ST 驱动器的接口管脚分配图。

为了保持您的主机与驱动器之间的通讯畅通，您需要这样来配置您的主机串行端口：9600bps、8 个数据位、一个停止位、无奇偶校验，并非巧合，这些设置是 windows-based PC 机的 COM 口的默认设置。



SCL 的通讯协议非常简单，所有的通讯都是由主机发起，只有在驱动器上电的时候，驱动器会持续发出通讯信号。在上电的时候驱动器发送“上电签名”，这是 MOONS' 软件使用的一个简单的标识符，它告诉软件连接了哪款驱动器和驱动器的硬件版本。除此之外，所有的通讯都由主机发起。

主机发送给驱动器的指令包的基本结构是一个以回车键结束的文本流，文本流总是由指令本身和用于该指令的参数组成，回车键表明指令发送到驱动器。下面是基本语法结构：

XXAB<cr>

在上面的语法中，“XX”表示 SCL 的指令代码，它总是由两个大写字母组成，“A”表示两个可能的参数中的第一个，“B”表示第二个参数，1、2 参数的长度没有限定，可以是字母或数字，通常是可选的。

一旦驱动器收到<cr>,它将决定是否解析指令。如果它解析了指令，驱动器就会执行或存储指令。如果 Ack/Nak 被激活（见 PR 指令），驱动器还会发送一个特定的字符给主机。指令被执行的肯定字符是“%”，指令被存储的肯定字符是“*”，如果驱动器不能解析指令，它将不会有任何动作。如果 Ack/Nak 被激活，它将发送一个否定字符“？”，否定字符通常伴随着一个表示特定错误的数字代码，见 PR 指令的错误代码列表。

开始

为了能够尽快使用 SCL 语言运行您的 ST 驱动器，按照下列基本步骤进行。

步骤一：安装软件

您的 ST 步进电机驱动器里附带了一张 CD，CD 里面包括了 MOONS' 驱动器的用户使用手册和所有能用到的应用软件。您也可以登陆到 www.moons.com.cn 网站免费下载。

下载 ST Configurator 和 SCL Setup Utility 软件后，安装在您的 windows-based PC 机上。ST Configurator 用来设置您的驱动器为 SCL 模式，SCL Setup Utility 用来发送 SCL 指令。

注意：笔记本没有串行端口，您需要一个 USB 串口适配器或 PCMCIA 串口适配器。

步骤二：用 ST Configurator 设置您的 ST 驱动器

安装好软件后，请打开您的驱动器和电机包装，确认必须的设备和配件，包括：

- 一个 ST 驱动器
- 一个两相步进电机。MOONS' 为 MSST 系列驱动器推荐了一些电机，当然您也可以使用自己的电机。
- 与您的 ST 驱动器一起提供的传输电缆（一端是 9 针 COM 接口，另一端是 RJ-11 标准连接头）。
- 一个小一字螺丝刀（驱动器附带）。
- 一个 24 或 48V 直流电源

将驱动器与电机连接起来，然后用编程电缆连接驱动器和 PC 机，驱动器与主机的距离请不要超过 3 米，以免影响 RS-232 通讯。

注意：不要在驱动器通电状态下接电机引线，在连接或拆卸电机时确认电源没有打开或没有接到驱动器。

启动 ST Configurator，开启电源，如果当您启动 ST Configurator 时，电源已经开启了，请先关闭电源，然后再开启，确保软件识别了驱动器的“上电签名”。否则可能会给您带来麻烦。

点击 ST Configurator 主界面的左下角的“Motor”按钮，设置您的电机。可以参照 ST Configurator 的帮助文件里关于设置的详细说明来设置您的步进电机的细节。

每款 ST 驱动器可以在下列四种工作模式中的一种下运行：脉冲&方向模式、速度模式、多轴网络模式和 SCL 模式。点击 ST Configurator 里的“Motion&I/O”按钮，然后选择 SCL，打开 SCL 的设置窗口。



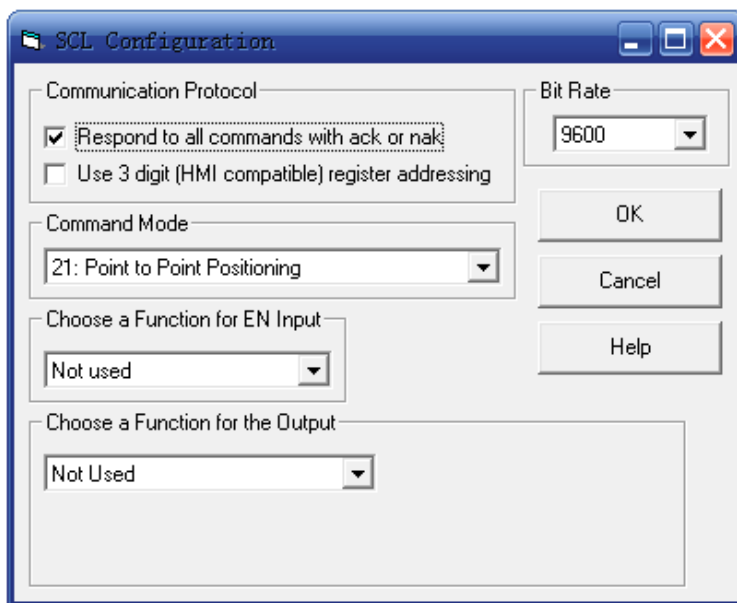
在 SCL 设置窗口里您可以做一些设置，它将影响在 SCL 模式下驱动器如何工作。查看下面的图片说明，了解这些设置。

通讯协议

通讯协议包括激活 Ack/Nak 和激活三位数字代码寄存器地址。激活 Ack/Nak 会使驱动器响应每一条指令，返回一个 Ack(%)或 Nak(*)的响应代码。三位数字代码寄存器地址是 HMI 的一个通用设置，因为大多数 HMI 使用数字代码寄存器地址，而字符寄存器地址用得较少。

上电模式

设置驱动器上电时的控制模式。驱动器可以在一些不同的工作模式下操作，而且仍然能用 SCL 指令通讯。大多数应用最初是从 CM21（点对点操作模式）开始。在 CM 指令的详细信息资料里可以更多的信息。



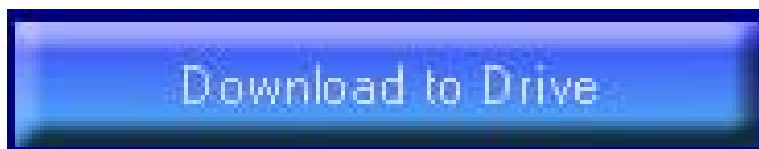
定义 EN 输入

允许您定义在您的应用中如何使用 EN 数字输入。EN 输入的功能选项有：驱动器使能输入，报警复位输入，或通用输入。

定义 OUT 输出

允许您定义如何使用驱动器的 OUT 数字输出，OUT 数字输出的功能选项有：报错输出，刹车输出，运动输出，转速输出，或通用输出。

如果您在 ST Configurator 里面设置好了电机并为驱动器选择了工作模式，点击主界面里的“Download to drive”按钮，并退出 ST Configurator。



步骤三：熟悉 SCL 指令

我们极力推荐您使用 SCL Setup Utility 软件来熟悉 SCL 指令。您可以到 www.moons.com.cn 下载 SCL Setup Utility。

启动 SCL Setup Utility 并连接驱动器到 PC 机(如果没有完成步骤二)。发送 SCL 指令很简单，在 SCL Setup Utility 的指令行输入指令，然后按 ENTER 键发送（所有的指令都是大写字母，所以开始时按 Caps Lock 键是个好办法）。按 ENTER 键完成两件事：它以<cr>字符结束指令，并自动发送完整的指令。请尝试下面的例子，在这些例子中，注意：<ENTER>表示按键盘上的 ENTER 键，就是类似于用<cr>字符结束指令。

注意：我们建议在电机轴不带负载的情况下练习 SCL 指令。在启动过程中您需要让电机轴自由旋转，以避免损坏您系统的机械结构。

AC25<ENTER>	设置加速度为 25 转/秒 ²
DE25<ENTER>	设置减速度为 25 转/秒 ²
VE5<ENTER>	设置速度为 5 转/秒
FL20000<ENTER>	电机顺时针运动 20000 步

如果您的电机在发送了 FL20000 之后没有运动。请查看驱动器上的 LED 是否有错误出现。如果出现，发送 AR 指令清除错误。如果清除错误后，LED 为固定绿色，，意味着驱动器被禁止。发送 ME 指令使能电机，然后确认 LED 是否为稳定的绿色闪烁状态。您可能仍然需要确认您在 ST Configurator 里的设置，确认电机电流是否设置妥当。然后重复上面的列子。

下面是另外一个例子，您可以试一下：

JA10<ENTER>	设置点动加速度为 10 转/秒 ²
JL10<ENTER>	设置点动减速度为 10 转/秒 ²
JS1<ENTER>	设置转速为 1 转/秒
CJ<ENTER>	启动点动
CS-1<ENTER>	改变点动为反方向 1 转/秒
SJ<ENTER>	停止点动

在上面的例子中，注意电机跳变到由 CS 确定的新速度。这个跳变受 JA 和 JL 指令的影响。用不同的 JA、JL、JS 和 CS 值试验上面的例子，观察他们是如何影响电机运动的。

步骤四：开发应用

对于不同的用户，这个部分包括了不同的内容。在到达这步之前，您可能要花足够的时间来用 SCL Setup Utility 熟悉 SCL 指令。

如果您的主机是PC机而您已经为您的应用做了一些必要的硬件配置，接下来将需要开发您PC机的应用软件，以适于发送SCL指令给您的驱动器。您可以选择使用什么软件或什么语言，例如VisualBasic, C++, LabView,或是其它的系统应用软件。MOONS'在自己的主页上提供了一些VisualBasic代码样本，展示如何从windows-based PC机应用软件发送SCL指令。您可以访问www.moons.com.cn下载。

如果您的主机是PLC，您需要按照简介里的管脚分布图和COM口设置列表连接并配置PLC上的ASCII模块、RS-232端口或Aux串行端口。这样您就可以通过敲PLC上的回车键来发送字符串。

如果您的主机是HMI，您需要像配置PLC那样来连接并配置HMI的串行端口，然后同样可以用正确的语法发送指令到驱动器。

指令

SCL 指令由两种基本类型：缓冲器指令和立即指令。缓冲器指令都被存储进指令缓冲器并在指令缓冲器外面执行。立即指令都不存储，接收到后立即执行。

缓冲器指令

缓冲器指令被存到指令缓冲器之后，在同一时间只能执行一条指令。如果您连续发送两条缓冲器指令到驱动器，例如发送 FL 指令后，接着发送 SS 指令，SS 指令则被存储在指令缓冲器里，等待直到 FL 指令完成后执行。在发送下一条指令前执行一条特殊的缓冲器指令，可以将指令缓冲器装满，这样可以不需要主机控制而有序的执行。特殊缓冲器指令，例如 PS、CT，使缓冲器存储并在期望的时间内暂停执行。

立即指令

立即指令都被立刻执行，如果需要的话，能与指令缓冲器并行执行。例如，当驱动器正在执行其它指令时，使用 BS 指令允许您检查缓冲器里的剩余空间。或使用 IS 指令查看数字输入端口的即时状态。立即指令被设计成当您需要的时候，可以在任何时间访问驱动器。这样可以让主机控制者可以有一个较高的频率从驱动器获取信息，通常用来确认电机位置或驱动器状态。

指令摘要

下面列表中所列指令都是使用 SCL 控制驱动器时可以用到的，每个表格有一些专栏，给出了各指令的相关信息。

- “指令”：由指令里的两个大写字母构成的指令代码；
- “指令描述”：每条指令的名称；
- “NV”：说明哪条指令是非易失性的：也就是说，当发送 SA 指令的时候，哪条指令被保存在非易失性的存储器里；
- “只写”和“只读”：当指令不是读/写兼容的时候两个中间选一个；
- “立即指令”：指明一个立即指令（其余的指令都是缓冲器指令）；
- “单位”：说明一些不同的事，如果指令有参数选项，则表示参数的单位，如果指令要求驱动器返回信息，则表示返回值的单位；
- “取值范围”：说明指令的参数的取值范围；
- “默认值”：说明存储在驱动器里的参数的默认值。

运动指令

指令	指令描述	非易失性	只写	只读	立即指令	单位	取值范围	默认值
AC	加速度	*				转/秒 ²	0.167~5461.167	100
AM	最大加速度	*				转/秒 ²	0.167~5461.167	1000
CJ	执行点动		*					
CS	改变点动速度				*	转/秒	+/-133.3333	-
DC	改变运行距离	*				步	0~2,147,483,647	0
DE	减速度	*				转/秒 ²	0.167~5461.167	100
DI	距离/位置	*				步	+/-2,147,483,647	20000
EG	细分	*				步/转	200~51200	20000
FC	改变速度的 FL		*					
FD	运动到第二个传感器		*					
FE	编码器跟随		*					
FL	运动到指定距离		*			步	+/-2,147,483,647	
FM	带屏蔽距离的 FS		*					
FO	带输出的 FL		*					
FP	运动到绝对位置		*			步	+/-2,147,483,647	
FS	运动到传感器信号有效		*					
FY	带安全距离的 FL		*					
HW	手轮							
JA	点动加速度	*				转/秒 ²	0.167~5461.167	100
JC	速度模式第二速度	*				转/秒	0.0042~133.3333	5
JD	点动禁止		*					
JE	点动使能		*					
JL	点动减速度	*				转/秒 ²	0.167~5461.167	100
JS	点动速度	*				转/秒	0.0042~133.3333	10
SH	找原点		*					
SJ	停止点动		*		*			
SK	停止并清除缓冲器		*		*	代码	D or none	
ST	停止		*		*			
VC	改变速度	*				转/秒	0.0042~133.3333	5
VE	速度	8				转/秒	0.0042~133.3333	10

配置指令

指令	指令描述	非易失性	只写	只读	立即指令	单位	取值范围	默认值
AI	报警复位输入	*				代码	1,2,3	3
AO	报警输出	*				代码	1,2,3	3
BD	刹车释放延时	*				秒	0~32.767	0.2
BE	刹车制动延时	*				秒	0~32.767	0.2
BO	刹车输出	*				代码	1,2,3	3
BR	波特率	*				代码	1~5	1
CC	改变电流	*				安培	0~I max	-
CD	待机电流延时	*				秒	0~32.767	0.4
CE	通讯错误			*	*	十六进制码	0~7F	-
CF	抗共振滤波器频率	*				赫兹	0~2000	241
CG	抗共振滤波器增益	*				整数	0~32767	12000
CI	待机电流	*				安培	0~I max	
CM	控制模式	*				代码	7,11~18,21,22	21
DA	定义地址	*				字符	见 DA 指令	1
DL	定义限位	*				代码	1,2,3	3
EG	细分	*				步/转	200~51200	20000
FI	输入端滤波器					输入/值	1,2,3/0~32767	
HG	四次谐波滤波器增益				*	整数	0~32767	
HP	四次谐波滤波器频率				*	整数	0~32767	
IA	即时模拟量反馈			*	*	输入	1 or none	
IC	即时电流反馈			*	*	安培		
ID	即时距离反馈			*	*	步		
IF	即时量格式				*	代码	H,D	H
IP	即时位置反馈			*	*	步		
IT	即时温度反馈			*	*	0.1℃		
IU	即时峰值电压反馈			*	*	0.1VDC		
IV	即时速度反馈			*	*	转/分		
MD	电机禁止		*					
ME	电机使能		*					
MO	运动输出	*				代码	1,2,3,4	3
PB	上电波特率	*				代码	1~5	1
PC	上电电流	*				安培	0~I max	
PI	上电待机电流	*				安培	0~I max	
PM	上电模式	*				代码	0~7	
PR	协议	*				整数	1~63	5
RE	重新开始		*		*			
SA	保存参数		*		*			
SF	步进滤波频率	*				赫兹	0~16000	2500
SI	使能输入	*				代码	1,2,3	3

SP	设置绝对位置	*				步	+/-2,147,483,647	
TD	传输延时	*				毫秒	0~32767	0

缓冲器指令

指令	指令描述	非易失性	只写	只读	立即指令	单位	取值范围	默认值
BS	缓冲器状态			*	*			
CT	继续		*		*			
PS	暂停		*					
SK	停止并清除缓冲器		*		*	代码	D or none	
SS	发送串		*					
WI	等待输入		*			输入/代码	1,2,3/L,H,F,R	
WT	等待时间		*			秒	0~320	

状态指令

指令	指令描述	非易失性	只写	只读	立即指令	单位	取值范围	默认值
AR	报警复位（立即）		*		*			
AX	报警复位（缓冲器）		*					
BS	缓冲器状态			*	*			
IC	即时电流反馈			*	*	安培		
ID	即时距离反馈			*	*	步		
IF	即时量格式				*	代码	H,D	H
IO	即时输出状态反馈				*	整数	0,1	
IP	即时位置反馈			*	*	步		
IT	即时温度反馈			*	*	0.1℃		
IU	即时电压反馈			*	*	0.1VDC		
IV	即时速度反馈			*	*	转/分		
MN	模式代码			*	*	代码	D,E	
RS	查询状态			*	*	代码	见 RS 指令	
RV	查询版本号			*	*			
SC	状态代码			*	*	十六进制	0000-FFFF	

I/O 指令

指令	指令描述	非易失性	只写	只读	立即指令	单位	取值范围	默认值
AD	模拟量死区	*				毫伏	0~255	50
AF	模拟量滤波器	*				整数	0~32767	14418
AG	模拟量速度增益	*				整数	0~32767	2400
AO	报警输出	*				代码	1,2,3	3
AP	模拟量位置增益	*				整数	0~32767	20000
AT	模拟量触发点	*				伏特	+/-10	2.5
AV	模拟量偏移	*				伏特	+/-10	0
AZ	模拟量零点（自动）		*					
BD	刹车释放延时	*				秒	0~32.767	0.2
BE	刹车制动延时	*				秒	0~32.767	0.2
BO	刹车输出	*				代码	1,2,3	3
FI	输入滤波器					输入/值	1,2,3/0~32767	
IH	立即输出高电平		*		*	输出口	1	
IL	立即输出低电平		*		*	输出口	1	
IO	即时输出状态反馈			*	*			
IS	即时输入状态反馈			*	*			
SI	使能输入	*				代码	1,2,3	3
SO	设置输出		*			输出口/状态	1/H,L	
WI	等待输入		*			输出口/状态	1,2,3/H,L,F,R	

寄存器指令

指令	指令描述	非易失性	只写	只读	立即指令	单位	取值范围	默认值
RL	寄存器加载（立即）				*			
RU	寄存器上传			*	*			

指令详细说明

这部分是一份按字母顺序排列的指令详细说明，包括您的驱动器能用到的所有指令。下面是对每页的信息的解释。

标题:	指令的两个大写字母组成的指令代码和指令名称;
描述:	解释指令的用途以及工作方式;
影响:	简单介绍了指令影响到的参数或其它指令;
参见:	有关联的指令;
指令说明:	说明指令的语法, 这行的格式总是两个字母的指令代码, 加上用到的一些参数, 不是所有的指令都有参数, 有些指令有可选的参数, 其它的指令则总是有一个参数, 可选的参数用{}表示, 必须的参数用()表示。
详细资料:	说明指令类型、指令用法和指令是否为非易失性, 当发送 SA 指令后, 非易失性指令会被存储, 通过 RL 指令, 这些指令可以发送数据到寄存器。寄存器地址将会在“Register Access”中给出, 同样也会给出指令的参数的详细资料。参数#1 和#2 给出了参数的简要说明。“units”显示驱动器是怎样理解参数的, “range”给出参数的取值范围, “default”说明参数的默认值。
举例:	举例说明如何使用这条指令。“command”下面是从主机控制器发送出来的指令串, 注意, 在这些例子中, 指令串后面没有给出<cr>, 但是任然需要它来结束这条指令串。 “Drive Sends”下面是驱动器的响应: 没有返回值用“-”表示。尽管如此, 如果 Ack 和 Nak 被启动, 每条指令发送之后, 都会有一个响应。“comments”说明指令发送后的其它信息。

注意: 在 ST 驱动器里使用数字输入或输出时, 记住指定 low 和 high 是非常重要的, 如果电流流过输入端或输出端, 即这个回路已经通电, 这种逻辑状态被定义为 low 或闭合。如果没有电流流过, 即回路没有通电, 或者输入输出端口没有连接, 这种逻辑状态定义为 high 或断开。在影响输入/输出的指令的参数里面, low 状态用“L”表示, high 状态用“H”表示, 例如: WI3L 表示“等待输入端 3 为 low”, SO1H 表示“输出端 1 为 high”。

AC-加速度

设置或查询所有点对点运动的加速度，单位转/秒²

影响：FC、FD、FM、FS、FP、FY、SH

参见：AM、DE、DI、VE。

指令说明：

AC{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“A”（017）
参数#1	加速度
-单位	转/秒 ²
-取值范围	0.167~5461.167
-默认值	100

举例

指令	驱动器返回	注释
AC100	-	设置加速度为 100 转/秒 ²
AC	AC=100	
AC25	-	设置加速度为 25 转/秒 ²
DE25	-	设置减速度为 25 转/秒 ²
VE1.5	-	设置速度为 1.5 转/秒
FL20000	-	顺时针运动 20000 步

AD-模拟量死区

设置或查询模拟量死区值，单位毫伏。模拟量死区值是模拟量输入的零点值周围的一个区域，模拟量输入的值在这个区域里时，被当作输入为零。这个零点可以被作为模拟量速度模式里零速度点来使用，或作为模拟量位置模式里的零位置。死区值是一个绝对值，在零点的正负方向都有效。

影响：模拟量输入

参见：CM 指令

指令说明：

AD{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	模拟量死区值
-单位	毫伏
-取值范围	0~255
-默认值	

举例

指令	驱动器返回	注释
AD100	-	设置模拟量死区值为 0.1V

AF-模拟量滤波器

给模拟量输入应用一个数字滤波器，这是一个复制模拟量输入的简单的单极滤波器，AF 指令的滤波器值关系到模拟量滤波器的期望值，单位赫兹，关系式如下：

$$\text{滤波器值} = 72090 / [(1400/x) + 2.2]$$

x: 模拟量滤波器的期望值，单位赫兹。

影响：模拟量输入

参见：IA、CM。

指令说明：

AF{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	滤波器值
-单位	整数
-取值范围	0~32767
-默认值	14418(500 赫兹)

*AF 值 28271 等于 4000.425 赫兹，AF 的值超过 28271 时它对模拟量滤波器的作用会被忽略。滤波器最大值约等于 4000 赫兹。

举例

指令	驱动器返回	注释
AF5000	-	设置模拟量输入带宽为 114.585 赫兹
AF	AF=5000	

AG-模拟量速度增益

设置或查询模拟量速度模式的增益值，增益值建立了模拟量输入和电机转速之间的关系，单位：0.25 转/分，例如，如果增益设置为 2400，当模拟量输入为 5V 时，电机转速为 10 转/秒。技巧：设置模拟量速度增益期望值的方法是，若转速单位为转/秒，则用电机转速期望值乘以 240，若转速单位为转/分，则乘以 4。

影响：模拟量速度模式

参见：CM 指令

指令说明：

AG{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	速度增益
-单位	0.25 转/分
-取值范围	0~32767
-默认值	2400

举例

指令	驱动器返回	注释
AG3000 -		设置模拟量速度模式最高速度为 12.5 转/秒
AG	AG=3000	

AI-报警复位输入

定义 EN 输入为报警复位输入。

在一次应用中，EN 输入只能指定一种定义，如果您想将 EN 输入用作报警复位输入，有以下两种方法可以定义：用 ST Configurator 软件或 AI 指令。如果控制模式为 CM13、14、17 和 18，AI 将不起作用，因为这些模式将 EN 输入作为速度改变输入，且优先级高于 AI 指令。同样，在设置 AI 指令后设置 SI 指令，会将 EN 输入重新指定为驱动器使能输入，并自动关闭任何报警输入用法（AI3）。换句话说，AI 指令、SI 指令和控制模式 CM13、13、14、17 和 18，各自给 EN 输入定义一种用法，专有的使用 EN 输入。控制模式有最高优先级，且总是不断的防止 AI 指令和 SI 使用 EN 输入，AI 指令和 SI 指令用覆盖写入的方式排斥另一个使用 EN 输入。

这里有三个 AI 指令定义的报警复位状态：

AI1：当 EN 输入端闭合（通电）时报警复位激活。这是一个边沿触发，如果当报警激活时开关已经闭合，则不会复位。输入端必须先断开然后再闭合来重置警报。

AI2：当 EN 输入端断开（断电）时报警复位激活。这是一个边沿触发，如果当报警激活时开关已经断开，则不会复位，输入端必须先闭合然后再断开来重置警报。

AI3：EN 输入不用于报警复位，可以作为一个通用输入口来使用。如果设置 AI 后，CM 设置为 13、14、17 和 18，或 SI 设置为 1 或 2，则 AI 被自动设置为 3。

指令说明：

AI{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	使用状态
-单位	整数
-取值范围	1,2,3
-默认值	3

举例

指令	驱动器返回	注释
AI1	-	EN 输入端闭合是复位所有可能的报警
AI	AI=1	

AL-报警代码

返回一个十六进制的报警代码值，这条指令对于检查驱动器串口是否有报警出现非常有用。

参见：AR 指令。

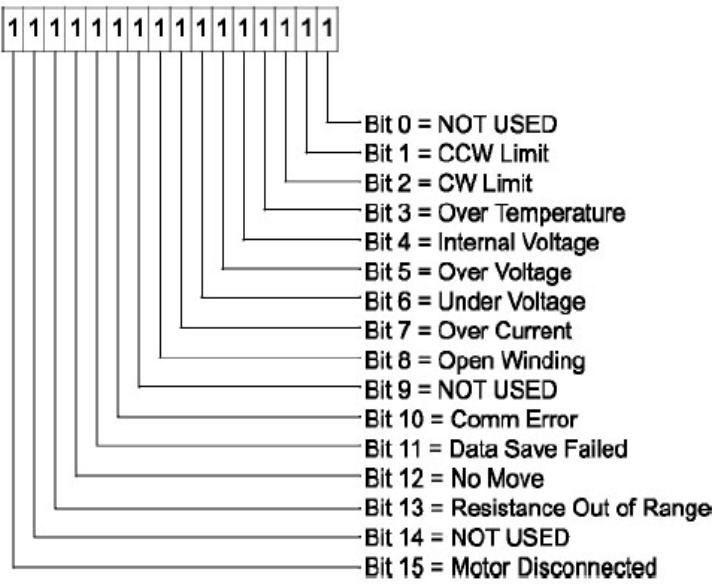
指令说明：

AL

详细说明：

指令类型	缓冲器指令
用法	只读
非易失性	不是
寄存器地址	“f” (054)

-单位	十六进制数
-----	-------



报警描述	位	Hex
不使用	0	0001
反转限位	1	0002
正转限位	2	0004
过温	3	0008
内电压	4	0010
过压	5	0020
欠压	6	0040
过流	7	0080
绕组开路	8	0100
不使用	9	0200
通讯错误	10	0400
数据保存失败	11	0800
没有动作	12	1000
电阻不匹配	13	2000
不使用	14	4000
电机连接不良	15	8000

举例

指令	驱动器返回	注释
AL	AL0000	没有报警
AL	AL0002	反转限位
AL	AL2001	反转限位，电阻不匹配

AM-最大加速度/减速度

设置或查询模拟量速度模式下的最大加速度/减速度。同时，在所有“Feed”模式的限位被激活或 ST、SK 指令激活时，作为减速度使用。

影响：ST、SK 指令，模拟量速度模式

参见：VM 指令

指令说明：

AM{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	最大加速度/减速度
-单位	转/秒 ²
-取值范围	0.167~5461.167
-默认值	

举例

指令	驱动器返回	注释
AM2000 -		设置最大加速度/减速度为 2000 转/秒 ²
AM	AM=2000	

A0-报警输出

定义驱动器数字输出为报警输出。

ST 驱动器的数字输出有四种定义：报警输出、刹车输出、运动输出或转速输出。这些定义都排斥其它的定义使用这个输出。所以只能定义一种。有两个方法来定义输出：通过ST Configurator软件或SCL 指令。设置输出为报警输出，使用 AO 指令和下列中的一个代码。

注意：设置 AO 指令参数为 1 或 2，将覆盖先前对输出口的定义，类似的，如果您在设置 AO 指令为 1 或 2 之后使用 BO 或 MO 指令来定义输出口，输出口的用法将被重新定义，并自动设置 AO 指令的参数为 3。

AO 指令可以定义以下三种输出状态：

AO1：报警出现时输出口闭合（通电）

AO2：报警出现时输出口断开（断电）

AO3：输出口不作为报警输出使用，可以定义为其它的自动输出口或通用输出口使用。

影响：数字输出口的定義

参见：AI、SI、BO、MO 指令

指令说明：

AO{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	输出状态
-单位	整数代码
-取值范围	1, 2, 3
-默认值	3

举例

指令	驱动器返回	注释
AO1	-	当报警或错误出现时，输出口闭合
AO	AO=1	

AP-模拟量位置增益

设置或查询模拟量位置增益，当驱动器为模拟量位置模式时，关系到电机的位置。增益值设定了当模拟量输入为满刻度值时的位置。可以用 ST Configurator 软件来设置模拟量输入的期望输入类型、刻度和偏移量。

影响：CM22（模拟量位置指令模式）

参见：AF、AZ 和 CM 指令

指令说明：

AP{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“x”（040）
参数#1	模拟量位置增益值
-单位	步
-取值范围	0~32767
-默认值	20000

举例

指令	驱动器返回	注释
AP8000	-	模拟量输入超过满刻度时的位置为 8000 步
AP	AP=8000	

AR-报警复位（立即）

复位报警并清除错误（如果有错误），如果错误或报警状态仍然持续，则报警没有被清除。

影响：报警代码

参见：AL、ME 和 MD 指令

指令说明：

AR

详细说明：

指令类型	立即指令
用法	只写
非易失性	不是
寄存器地址	无

举例

指令	驱动器返回	注释
AR	-	报警被清除

AT-模拟量死区

设置或查询 AIN 输入口的模拟量输入死区，它用于 FS 指令。死区值设定一个模拟电压来确定传感器状态或触发值。

影响：所有 FS 类型指令

参见：AF,AZ,FS,FM 和 FY 指令

指令说明：

AT{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“Y”(041)
参数#1	模拟量死区值
-单位	伏
-取值范围	-10~10
-默认值	2.5

举例

指令	驱动器返回	注释
AT5	-	设置模拟量死区值为 5 伏
AT	AT=5	

AV-模拟量偏移量值

设置或查询模拟量输入的模拟量偏移值。

影响：所有模拟量输入定义

参见：AF,AP,AZ,CM 指令和“Feed”指令

指令说明：

AV{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“Z”(042)
参数#1	模拟量偏移值
-单位	伏
-取值范围	-10~10
-默认值	0

举例

指令	驱动器返回	注释
AV0.25	-	设置模拟量偏移值为 0.25 伏
AV	AV=0.25	

AX-模拟量复位（缓冲器）

复位驱动器的错误并清除报警代码。这条指令类似于 AR 指令，但是这是缓冲器指令，而不是立即指令。

影响：报警代码

参见：ME 和 AR 指令

指令说明：

AX

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无

举例

指令	驱动器返回	注释
AX	-	复位驱动器错误并清除报警代码

AZ-模拟量零点

激活模拟量“auto offset”运算法则。这个运算法则同样可以在 ST Configurator 软件里激活。这条指令对定义模拟量输入当前的电压为零点或偏移值时非常有用。

影响：所有模拟量输入的定义

参见：AF,AP,AVCM 指令和“Feed”指令

指令说明：

AZ

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	不是
寄存器地址	无

举例

指令	驱动器返回	注释
AZ	-	启动模拟量自动偏移运算法则

例如：在驱动器的 AIN 和 GND 端加载 1VDC 电压，然后发送 AZ 指令，然后加载 4VDC 电压，发送 IA 指令，返回值将非常接近 IA=3.00

BD-刹车释放延时

这条指令只有在 BO 指令设置为 1 或 2 时才有效。当驱动器使能后，这个时间就是等待刹车释放的延时，在动作开始之前，必须等待延时结束。当驱动器使能且设置了刹车输出后，延时计数器马上开始计数。BD 指令设置毫秒级的延时。

影响：所有“Feed”和“Jog”指令

参见：BE 和 BO 指令

指令说明：

BD{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	延时时间
-单位	秒
-取值范围	0~32.767
-默认值	0.2

举例

指令	驱动器返回	注释
BD0.2	-	设置刹车释放延时为 200 毫秒
BD	BD=0.2	

BE-刹车使能延时

这条指令只有在 BO 指令设置为 1 或 2 时才有效，当驱动器禁止后，这个时间是驱动器输出的实际禁止的延时时间。当使用数字输出口为刹车输出口时，输出口与禁止指令同时立即被激活，然后驱动器等待延时结束再关闭驱动器输出。

影响：所有“Feed”和“Jog”指令

参见：BD 和 BO 指令

指令说明：

BE{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	延时时间
-单位	秒
-取值范围	0~32.767
-默认值	0.2

举例

指令	驱动器返回	注释
BE0.25	-	设置刹车使能延时为 250 毫秒
BE	BE=0.25	

BO-刹车输出

定义驱动器数字输出口为刹车输出口。

ST 驱动器的输出口有四种定义：报警输出、刹车输出、运动输出或转速输出。这些定义都排斥其它的定义使用这个输出。所以只能定义一种。有两个方法来定义输出：通过ST Configurator软件或 SCL 指令。设置输出为刹车输出，使用 BO 指令和下列中的一个代码。

注意：设置 BO 指令参数为 1 或 2，将覆盖先前对输出口的的定义，类似的，如果您在设置 BO 指令为 1 或 2 之后使用 AO 或 MO 指令来定义输出口，输出口的用法将被重新定义，并自动设置 BO 指令的参数为 3。

BO 指令可以定义以下三种输出状态：

BO1：当驱动器被使能时输出口闭合（通电），禁止时被断开；

BO2：当驱动器被禁止时输出口断开（断电），使能时被闭合；

BO3：输出口不作为刹车输出使用，可以定义为其它的自动输出口或通用输出口使用。

影响：数字输出口的定義

参见：AI,AO,ME,MD,MO,SI 指令

指令说明：

BO{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	输出状态
-单位	整数
-取值范围	1, 2, 3
-默认值	3

举例

指令	驱动器返回	注释
BO1	-	当驱动器使能时，输出口闭合（通电）
BO	BO=1	

BR-波特率

设置或查询串行通讯的波特率

当驱动器上电时默认为 9600bps，如果主机系统在一秒钟内没有检测到，且驱动器被设置为 SCL 模式，驱动器将根据存储在寄存器里的参数来设置波特率，主机可以在任何时间用这条指令设置波特率。

注意：波特率设置后将立即生效。

影响：与主机之间的串行通讯

参见：TD 和 PR 指令

指令说明：

BR{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	波特率
-单位	整数代码
-取值范围	1=9600 2=19200 3=38400 5=57600 5=115200
-默认值	1

举例

指令	驱动器返回	注释
BR2	-	波特率立即被设置为 19200bps.
BR	BR=2	

BS-缓冲器状态

查询驱动器的指令缓冲器里可以用的指令空间的数量。这项技术非常方便，只需要发送一条指令就能知道是否有足够的空间存放指令，而不需要计算。驱动器的返回值只要大于“1”，就可以发送一条指令。如果驱动器返回“63”，说明缓冲器是空的，如果返回值为“0”，则不能存放更多的缓冲器指令。

指令说明：

BS

详细说明：

指令类型	立即指令
用法	只读
非易失性	不是
寄存器地址	无
-单位	缓冲器里的空间

举例

指令	驱动器返回	注释
BS	BS=20	缓冲器还可以存储 20 条指令。

CC-改变电流

设置或查询驱动器电流。

影响：所有“Feed”指令，CJ,SH,WI（点动）和力矩模式。

参见：PC 指令

指令说明：

CC{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“N”（030）
参数#1	电机电流
-单位	安培
-取值范围	0~5（MSST5-S） 0~10（MSST10-S）

举例

指令	驱动器返回	注释
CC4.5	-	设置电机电流为 4.5 安培
CC	CC=4.5	

CD-待机电流延时时间

设置和查询驱动器从运行电流跳变到待机电流的延时时间。延时从步进电机完成最后一步后开始。在每个脉冲/方向模式里，驱动器在每接收到一个脉冲后都会复位待机电流延时。

指令说明：

CD{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	延时时间
-单位	秒
-取值范围	0~32.767
-默认值	0.4

举例

指令	驱动器返回	注释
CD0.2	-	设置待机电流延时为 200 毫秒
CD	CD=0.2	

CE-通讯错误

读取通讯错误代码。这条指令可以读取当前出现的通讯错误的代码。返回值是一个十六进制码，下面表格中是每一位的定义的详细资料。

指令说明：

CE

详细说明：

指令类型	立即指令
用法	只读
非易失性	不是
寄存器地址	无
参数#1	通讯错误代码
-单位	十六进制码
-取值范围	Bit0=奇偶校验位（不使用） Bit1=成帧误差 Bit2=噪音干扰 Bit3=溢出错误 Bit4=接收缓冲器已满 Bit5=发送缓冲器已满 Bit6= bad SPI op-code

举例

指令	驱动器返回	注释
CE	CE=10	接收寄存器已满

CF-抗共振滤波器频率

设置和查询抗共振滤波器频率，单位赫兹。这条指令与抗共振滤波器增益（CG）联合使用，以消除中频共振。

注意：我们强烈建议使用 ST -Configurator 软件来设置，在 ST --Configurator 软件的电机设置窗口精确设置电机参数及负载转动惯量，软件将自动计算出 CF 和 CG。

影响：电机中频性能

参见：CG 指令

指令说明：

CF{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	滤波器频率
-单位	赫兹
-取值范围	0~2000
-默认值	241

举例

指令	驱动器返回	注释
CF1400	-	设置抗共振滤波器频率为 1400 赫兹
CF	CF=1400	

CG-抗共振滤波器增益

设置或查询抗共振滤波器增益，单位赫兹。这条指令与抗共振滤波器频率（CF）联合使用，以消除电机的中频共振。

注意：我们强烈建议使用 *ST-Configurator* 软件来设置，在 *ST-Configurator* 软件的电机设置窗口精确设置电机参数及负载转动惯量，软件将自动计算出 CF 和 CG。

影响：电机的中频性能

参见：CF 指令

指令说明：

CF{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	滤波器增益
-单位	整数
-取值范围	0~32767
-默认值	12000

举例

指令	驱动器返回	注释
CG800	-	设置抗共振滤波器增益为 800
CG	CG=800	

CI-改变待机电流

待机电流是当电机停止运动时电机每相电流的标准值。待机电流标准低于运行电流标准有助于电机冷却。通常使用的待机电流标准是运行电流标准的 50%。电机运动完最后一步后，有一个延时，延时结束后待机电流才有效。延时时间由 CD 指令设置。

影响： 电机待机时的电流

参见： CC,PI,CD

指令说明：

CI{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“O”(031)
参数#1	电流
-单位	安培
-取值范围	0~5(MSST5-S) 0~10(MSST10-S)

举例

指令	驱动器返回	注释
CI2	-	设置待机电流为 2A
CI	CI=2	

CJ-开始点动

电机开始点动。电机以点动加速度加速到点动速度，直到停止。若要停止点动，使用停止点动指令 SJ，以点动减速度（点动减速度由 JL 设定）停止。若要迅速停止点动，使用 ST 指令（最大减速度由 AM 设定）。但是如果速度太快或者负载惯量过高，电机可能失步。DI 指令设置点动方向，CS 指令改变点动速度和方向。CS 不会影响到 JS。

参见：CS,DI,JA,JL,JS,SJ 和 ST 指令。

指令说明：

CJ

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无

举例

指令	驱动器返回	注释
JS5	-	设置点动速度为 5 转/秒
CJ	-	开始点动
CS10	-	改变点动速度为 10 转/秒
SJ	-	停止点动

CM-控制模式

设置或查询驱动器的控制模式。Feed 指令和 Jog 指令在执行前自动设置合适的控制模式。使用 ST-Configurator 软件可以进行更多的控制模式的自动设置。

警告：随意改变控制模式可能导致电机高速运行或其它意想不到的结果。

指令说明：

CM{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	控制模式
-单位	整数代码
-取值范围	7=脉冲&方向模式 11=模拟量速度模式 12~18=（见下表） 21=点对点模式 22=模拟量位置模式
-默认值	21

控制模式 12~18:

注意：建议使用 ST Configurator 软件运行速度模式。

12-模拟量速度模式，STEP 输入作为运动/停止输入；

13-模拟量速度模式，EN 输入作为速度变化输入；

14-模拟量速度模式，STEP 输入作为运动/停止输入，EN 输入作为速度变化输入；

15-速度模式（JS 设置速度）；

16-速度模式（JS 设置速度），STEP 输入作为运动/停止输入；

17-速度模式（JS 设置速度），EN 输入作为速度变化输入；

18-速度模式（JS 设置速度），STEP 输入作为运动/停止输入，EN 输入作为速度变化输入；

举例

指令	驱动器返回	注释
CM7	-	设置驱动器为脉冲&方向模式
CM	CM=7	

CS-改变速度

设置或改变点动时的点动速度，单位转/秒。在点动运动时，可以用 CS 指令来改变点动的速度。CS 指令的参数可以是正数也可以是负数，以改变点动方向。变化的加速度或减速度由 JA 或 JL 指令设定。CS 指令不会影响到 JS 或 DI。

影响： 点动时的点动速度。

参见： CJ 和 JS 指令

指令说明：

CS{参数#1}

详细说明：

指令类型	立即指令
用法	读/写
非易失性	是
寄存器地址	“J”(026)
参数#1	点动速度
-单位	转/秒
-取值范围	-133.3333~133.3333 “-”表示 CCW，无符号表示 CW

举例

指令	驱动器返回	注释
CS2.5	-	改变点动速度为 2.5 转/秒
CS	CS=2.5	查询当前点动速度
CS-5	-	改变点动速度为 5 转/秒,方向逆时针。

CT-继续

重新执行由于 PS 指令中止的缓冲器指令。PS 指令可以暂停执行指令缓冲器里的指令。且 PS 指令执行后，后面发送的指令都被存储到指令缓冲器，直到发送 CT 指令，或指令缓冲器以满，才继续执行收到的指令。

参见： PS,ST 和 SK 指令

指令说明：

CT{参数#1}

详细说明：

指令类型	立即指令
用法	只写
非易失性	不是
寄存器地址	无

举例

指令	驱动器返回	注释
CT	-	继续执行暂停的指令

DA-定义地址

设置 ST 驱动器的地址。因为 ST 驱动器只工作在点对点的 RS-232 通讯协议，所以驱动器地址通常不需要去改变默认值“1”。

参见： PR 指令

指令说明：

DA{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	RS-485 网络地址
-单位	字符
-取值范围	有效的地址字符： !"#\$%&`()*+,-./0123456789;:<>?
-默认值	1

举例

指令	驱动器返回	注释
DA2	-	设置驱动器地址为“2”
DA	DA=2	
SA	-	保存新地址到非易失性寄存器

DC-改变距离

设置或查询“改变距离”或偏置距离，单位 step。DC 指令用于各种各样的运动指令，以多定义一个距离参数。所有的运动指令的许多时候都使用 DI 指令，有一些则需要 DC，例如 FC、FM、FO 和 FY，这些指令在 DC 距离结束后执行指令里可变的动作。例如：FM 类似于 FS，但是在 FM 指令里，在 DC 距离结束之前传感器输入被忽略，这对屏蔽某些开关或传感器触发非常有利。因为 DI 指令的符号决定方向，所以 DC 不改变方向。

影响： FC、FY、FO 和 FM 指令

参见： VC 指令

指令说明：

DC{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“C”(019)
参数#1	距离
-单位	步
-取值范围	0~2, 147, 483, 647
-默认值	0

举例

指令	驱动器返回	注释
DC80000-		设置改变距离为 80000 步
DC	DC=80000	
DI-100000	-	设置全程距离为 100000 步，方向 CCW
DC50000	-	设置改变距离为 50000 步
VE5	-	设置速度为 5 转/秒
VC2	-	设置改变速度为 2 转/秒
FC	-	执行 FC 指令

DE-减速度

设置或查询所有“F”指令用到的减速度，单位转/秒²

影响： FC,FD,FE,FL,FM,FS,FP,FY,SH 指令

参见： AC,AM,DE,DI,VE 指令

指令说明：

DE{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“B”(018)
参数#1	减速度
-单位	转/秒 ²
-取值范围	0.167~5461.167
-默认值	100

举例

指令	驱动器返回	注释
DE125	-	设置减速度为 125 转/秒 ²
DE	DE=125	
AC25	-	设置加速度为 25 转/秒 ²
DE10	-	设置减速度为 10 转/秒 ²
VE5	-	设置速度为 5 转/秒
FL20000	-	执行 FL 指令，行程 20000 步，方向顺时针

DI-距离/位置

设置或查询运动距离，单位 step。DI 指令的符号决定运动方向：“-”表示 CCW，无符号表示 CW。DI 既用于相对运动，也用于绝对运动，FL 指令就是一个相对运动的例子。FP 指令是一个绝对运动，在 FP 指令里，DI 指令设置的是绝对位置，而不是相对位置。

影响：所有的“Feed”指令和 SH 指令

参见：AC,DC,DE 和 VE 指令

指令说明：

DI{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“D”(020)
参数#1	距离
-单位	步
-取值范围	-2,147,483,647~2,147,483,647 “-”表示 CCW，无符号表示 CW
-默认值	20000

举例

指令	驱动器返回	注释
DI20000 -		设置全程距离为 20000 步，方向 CW
DI	DI=20000	
DI-8000 -		设置全程距离为 8000 步，方向 CCW
FL -		开始 FL 指令
SP0 -		设置当前位置为绝对零位置
DI20000 -		设置绝对位置为 20000 步，方向 CW
FP -		开始运动到设定的绝对位置
DI10000 -		设置绝对位置为 10000 步，方向 CW
FP -		开始运动到设定的绝对位置，方向 CCW

DL-定义限位

定义 STEP 输入为正转限位输入，DIR 输入为反转限位输入。这两个输入一起被定义为限位输入，而且是同样的连接类型。定义为限位后，如果其中一个输入被触发，电机将停止在那个方向的运动，且驱动器的状态 LED 灯将显示报警代码。2 绿 2 红表示正转限位，1 绿 2 红表示反转限位。

在一次应用中，STEP 和 DIR 输入只能指定一种定义，如果您想将 STEP 和 DIR 输入作为限位输入，可以用这样两种方法：用 ST Configurator 软件或用 DL 指令。如果驱动器控制模式为 CM7，11，12，13，14，15，16，17 或 18，DL 指令将无效，因为这些模式已经定义了输入口，并且优先级高于 DL 指令。同样，在设置 DL 指令后再设置 JE 指令，将会重新定义 STEP 和 DIR 输入为点动输入，并取消任何限位用法。同样，控制模式 CM7，11，12，13，14，15，16，17 或 18 也会重新定义 STEP 和 DIR 输入。他们各自排斥其它的使用这两个输入。控制模式拥有最高优先级，且总是防止 DL 和 JE 使用这两个输入，DL 和 JE 则按覆盖写入的原则排斥另一个使用。

DL 指令可给限位定义下列三种状态：

DL1：当输入口闭合（通电）时限位有效，电机自动以 AM 指令减速。

DL2：当输入口断开（断电）时限位有效，电机自动以 AM 指令减速。

DL3：STEP 和 DIR 输入不作为限位输入，可以定义为其它通用输入口。当控制模式设置为 CM7，11，12，13，14，15，16，17，18 或在 DL 之后定义 JE，DL 将被自动设置为 DL3。

影响： 所有“F”指令，CJ,SH,WI（在点动时）指令

参见： AM 指令

指令说明：

DL{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	限位输入状态
-单位	整数
-取值范围	1， 2， 3
-默认值	3

举例

指令	驱动器返回	注释
DL1	-	输入口闭合时限位触发。电机减速度由 AM 设定。
DL	DL=1	
DL3	-	输入口为通用输入口。电机减速度由 AM 设定。

EG-细分

设置或查询步进电机的步进细分。在大多数应用中，步进细分在 ST Configurator 软件里设置好后不再改动。虽然如此，一些应用可能需要频繁的改变细分。例如：当使用编码器跟随模式时通常需要改变细分。因此，EG 指令设置电机对主编码器的跟随细分。为了改变跟随细分，必须在执行 FE 指令之前设置 EG 指令。同样，也可以用 RL 指令通过改变“R”寄存器的值来有效改变跟随细分。

注意：EG 的值时“R”寄存器值的两倍，换句话说，如果您设置 EG 为 10000，“R”寄存器的值就是 5000。这里显示的 EG 指令和“R”寄存器的值是基于 1.8 度步进电机的。

影响： 控制模式 CM7，FE 和 HW 指令

参见： “R”寄存器，CM,FE,HW 和 RL 指令

指令说明：

EG{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“R”(034)
参数#1	步进细分
-单位	步/转
-取值范围	200~51200
-默认值	20000

举例

指令	驱动器返回	注释
EG20000 -		设置细分为 20000 步/转

FC-改变速度的 FL

执行一个改变速度的 FL 指令。DI 指令设置全程距离和方向，AC 和 DE 指令设置加速度和减速度。初始速度为 VE，当运动完 DC 设置步数后，速度变为 VC 设定数值。

注意：如果 DC 大于 DI 减去减速距离，速度将不会变化。

参见：VC 和 DC 指令

指令说明：

FC

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	是
寄存器地址	无

举例

指令	驱动器返回	注释
DI50000	-	设置全程距离为 50000 步
VE	-	设置初始速度为 5 转/秒
DC40000	-	设置距离为 40000 时改变速度
VC0.5	-	设置改变速度为 0.5 转/秒
FC	-	开始运动

FD-运动到第二个传感器

开始电机以 AC 加速度加速到 VE，当到达第一个传感器后，电机以 DE 减速度减速到 VC。当到达第二个传感器后，电机以 DE 减速度减速到停止。DI 指令确定方向和停止时的减速距离，模拟量和数字输入都可以用于这条指令。

参见：FM,FS,FY 和 VC 指令；见 AT 指令里 AIN 口用于传感器输入的部分。

指令说明：

FD(参数#1)(参数#2)

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	第一个输入口编号，输入状态
-单位	整数，字母
-取值范围	整数：1=STEP,2=DIR,3=EN,4=AIN 字母：L=low,H=high F=fallin edge,R=rising edge
参数#2	第二个输入口编号，输入状态
-单位	整数，字母
-取值范围	整数：1=STEP,2=DIR,3=EN,4=AIN 字母：L=low,H=high F=fallin edge,R=rising edge

举例

指令	驱动器返回	注释
AC50	-	设置加速度为 50 转/秒 ²
DE50	-	设置减速度为 50 转/秒 ²
DI-1	-	设置方向为 CCW
VE5	-	设置初始速度为 5 转/秒
VC1	-	设置改变速度为 1 转/秒
FD1F2H	-	开始 FD 运动：1 输入口下降沿出现时减速到 VE,2 输入口变为高电平时减速到停止。

注意：当使用模拟量输入口时，“L”表示输入值低于 AT 指令值，H 表示高于 AT 指令值。

FE-编码器跟随

设置驱动器为编码器跟随模式，直到给定的数字或模拟量信号出现。主编码器的 A 和 B 通道接到驱动器的 STEP 和 DIR 输入口。在执行 FE 指令前使用 EG 指令设置跟随细分，或使用“R”寄存器。当 FE 指令开始执行，驱动器将按 AC 加速度将速度变化到跟随速度，这样做防止电机加速过激。此外，驱动器还将跟随主编码器的方向，CW 和 CCW。一旦输入状态要求停止，电机只有运动在 DI 设定的方向上时会完全停止。这条指令同样也像 FS 指令工作，使从 FE 指令回到在这之前的运动模式时，当输入状态触发停止后，它会按照 DI 寄存器里的值运动一段距离。

注意：您必须在 ST Configurator 软件里设置 STEP 和 DIR 为编码器跟随输入，在脉冲&方向运动模式里选择 A/Bquadrature。在编码器跟随时改变“R”寄存器要非常小心，因为有些运动参数会溢出，然后运动会出现意想不到的情况。

参见：EG 指令和“R”寄存器；见 AT 指令里 AIN 输入用于传感器输入的部分。

指令说明：

FE{参数#1}

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	输入口，输入状态
-单位	整数，字母
-取值范围	整数：3=EN,4=AIN 字母：L=low, H=high F=falling edge, R=rising edge

举例

指令	驱动器返回	注释
AC500	-	限制编码器跟随模式里加速度为 500 转/秒 ²
DI8000	-	设置停止偏移距离为 8000 步
FE3L	-	跟随编码器，直到 EN 输入为低电平

FI-输入滤波器

应用一个数字滤波器到一个给定的输入。数字输入在 FI 指令的时间周期后和在输入状态被确认前必须保持在同一级别。例如，时间周期设置为 100，数字输入在 100 个机器周期（一个机器周期=250 毫秒）后和在高电平被确认前必须仍然保持高电平，滤波器值为“0”时滤波器被禁止。这条指令可以用于给 STEP、DIR 和 EN 提供滤波器。

影响： 所有使用输入的指令

参见： WI,FE,FD,FM,FS 和 FY 指令

指令说明：

FI(参数#1){参数#2}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	输入
-单位	整数
-取值范围	1=STEP,2=DIR,3=EN
参数#2	滤波器值
-单位	机器周期
-取值范围	0~32767

举例

指令	驱动器返回	注释
FI1100	-	要求 STEP 输入信号保持相同的状态 25 毫秒
FI1	FI1=100	

FL-运行到指定距离

执行运动到设定距离。DI 指令设置运动的距离和方向。VE, AC 和 DE 指令分别设定速度、加速度和减速度。可选的参数允许运行一个局部距离, 而不是 DI 的距离

注意: 如果 FL 自带了参数, DI 指令将被忽视。

参见: AC,DE,DI,VE 指令

指令说明:

FL{参数#1}

详细说明:

指令类型	缓冲器指令
用法	只写
非易失性	是
寄存器地址	无
参数#1	相对距离
-单位	步
-取值范围	-2,147,483,647~2,147,483,647 “-”表示 CCW, 无符号表示 CW

举例

指令	驱动器返回	注释
AC100	-	设置加速度为 100 转/秒 ²
DE150	-	设置减速度为 150 转/秒 ²
VE8	-	设置速度为 5 转/秒
DI20000	-	设置距离为 20000 步, 方向 CW
FL	-	开始运动到设定距离
FL10000	-	开始运动 10000 步, 方向 CW, DI 被忽略。
FL-400	-	开始运动 400 步, 方向 CCW, DI 被忽略。

FM-带屏蔽距离的 FS

执行一个 FS 指令，但是在运动开始的 DC 距离内传感器被忽略。对在感应到正确的停止信号前屏蔽开关或清除一段距离时非常有用。DI 设置为停止信号触发后的运动距离，AC 设置加速度，VE 设置速度，DE 设置减速度。

参见：DC 和 FS 指令；见 AT 指令里 AIN 输入用于传感器输入的部分。

指令说明：

FM{参数#1}

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	输入口，输入状态
-单位	整数，字母
-取值范围	整数：1=STEP,2=DIR,3=EN,4=AIN 字母：L=low,H=high F=fallin edge,R=rising edge

例如：在传送带上输送零件时，传感器检测到零件的前沿时停止。但如果零件上有洞（很多零件都有这样的情况），那么当试图输送下一个零件时你可能只移动了前一个零件一小段距离就停了下来。解决办法就是使用 FM 指令来代替 FS 并根据零件尺寸设置 DC。

例如：零件有 6 英寸长，机械传动链转速比 2000 步/英寸。5 英寸传送距离范围内屏蔽传感器信号，然后检测到传感器有效信号后 1 英寸停下来。完成此任务的指令序列如下：

指令	驱动器返回	注释
DI2000	-	设置通过传感器后的停止距离为 2000 步（1 英寸）
DC10200	-	设置屏蔽距离为 10200 步（5.1 英寸）
FM1F	-	开始 FM 运动，传感器连接到 STEP，下降沿有效

F0-带输出的 FL

与 FL 指令类似，但是在运动过程中改变输出状态。DI 指令定义全程距离，AC、DE 和 VE 设定加速度、减速度和速度，DC 定义改变输出状态的距离，DC 须小于 DI。

注意：在使用 FO 指令之前，专用的输出定义-报警输出，刹车输出，运动输出-必须设置为通用输出。见 AO、BO 和 MO 指令。

参见：DC, AO, BO 和 MO 指令

指令说明：

FO{参数#1}

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	输出口，输出口定义
-单位	整数，字母
-取值范围	整数：1 字母：L=low, H=high F=falling edge, R=rising edge

举例：

指令	驱动器返回	注释
AC100	-	设置加速度为 100 转/秒 ²
DE100	-	设置减速度为 100 转/秒 ²
VE2.5	-	设置速度为 2.5 转/秒
DI20000	-	设定全程距离为 20000 步
DC15000-	-	设定输出变化距离为 15000 步
FO1L	-	开始 FO 指令，距离为 15000 时输出为低电平

FP-运动到指定位置

执行一个 FP 运动。运动的位置由最后一个 DI 设定。VE、AC 和 DE 指令分别定义速度、加速度和减速度。可选的参数允许 FP 指令使用一个局部的绝对位置，而不是 DI 的值。

注意：如果 FP 指令使用自带的参数，则最后一个 DI 指令将被忽略。

参见：AC,DE,DI 和 VE 指令。

指令说明：

FP{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	不是
寄存器地址	无
参数#1	绝对位置
-单位	步
-取值范围	-2,147,483,647~2,147,483,647

例如，在电机回到原位后定义该位置为零点位置然后运动绝对距离 8000 步：

指令	驱动器返回	注释
SP0	-	设置当前位置为绝对零位置
VE5	-	设置速度为 5 转/秒
DI80000	-	设置运动位置为 80000 步
FP	-	开始运动到绝对位置 80000 步
FP40000	-	开始运动到绝对位置 40000 步，DI 无效

FS-运动到传感器信号有效

执行一个 FS 指令。电机运动直到传感器触发后，将停止在越过传感器的精确的位置。这个位置由 DI 指令值设置，DI 指令的符号确定运动方向（“-”表示 CCW，无符号表示 CW）。VE、AC 和 DE 分别设置速度、加速度和减速度。

电机运动在一定的速度，并有确定的减速度时，停止时需要一段距离来减速。如果距离过短，电机将越过预定的位置。使用下面的公式来计算最短距离，确定一个速度 V （转/秒）和减速度 D （转/秒²），和细分 R （步/转）。

$$\text{最小减速距离} = V^2 R / 2D$$

参见：FD, FM 和 FY 指令，见 AT 指令里 AIN 用于传感器输入的部分。

指令说明：

FS{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	不是
寄存器地址	无
参数#1	输入口，输入状态
-单位	整数，字母
-取值范围	整数：
-默认值	整数： 1=STEP, 2=DIR, 3=EN, 4=AIN 字母： L=low, H=high F=fallin edge, R=rising edge

举例

指令	驱动器返回	注释
AC100	-	设置加速度为 100 转/秒 ²
DE100	-	设置减速度为 100 转/秒 ²
DI1000	-	设置减速距离为 1000 步
VE0.5	-	设置速度为 0.5 转/秒
FS1L	-	开始 FS 运动，当 STEP 输入为低电平时停止。

注意：在使用 ST 驱动器数字输出和输入口的时候，您需要特别注意口设置为高或者低。如果有电流流入或者流出一个输入口或输出口的时候，也就是电流回路形成的时候，输入/输出口的逻辑状态被定义为低或者闭合。如果没有电流流动，也就是电流回路没有形成，或者输入/输出口没有连接，输入/输出口的逻辑状态为高或开路。逻辑低用 L 来表示。例如，WI3L 意思是 wait for input 3(EN 输入) 低，SO1L 意思是 set output 1 low。逻辑高用 H 表示。

提示：当使用 ST 驱动器模拟输入的时候，L 指定的模拟值会小于 AT 命令设置的值。同样的，H 指定的模拟值会高于 AT 命令设置的值。

FY-带安全距离的 FS

执行一个 FS 指令，但是有一个 DC 设置的安全距离。如果运动到安全距离时传感器还没有触发，电机将停止，并且驱动器发送一个叹号(!)给主机。DI 指令定义运动方向和传感器触发后的停止距离。VE、AC 和 DE 分别定义速度、加速度和减速度。注意：如果传感器没有触发，最终的位置将是安全距离减去负载需要的减速距离。

这条指令对消除机械碰撞或者检测标签卷结束的应用中是很有用的。例如，：您正在输送标签。你想让传感器检测到标签的前端后2000 步停下来，标签间距为60000 步。因此，如果在60000 步内检测不到有效的传感器信号，说明目前是在标签卷的尾端。

参见：DC,DE,FD,FM 和 FS 指令；见 ST 指令里 AIN 端口用于传感器输入的部分。

指令说明：

FY(参数#1)

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	输入口，输入状态
-单位	整数，字母
-取值范围	整数：1=STEP,2=DIR,3=EN,4=AIN 字母：L=low,H=high F=fallin edge,R=rising edge

举例

指令	驱动器返回	注释
AC50	-	设置加速度为 50 转/秒 ²
DE50	-	设置减速度为 50 转/秒 ²
VE2.5	-	设置速度为 2.5 转/秒
DI2000	-	设置越过传感器的距离为 2000 步，并确定方向为 CW
DC60000-	-	设置安全距离为 60000 步
FY2L	-	开始 FY 运动：安全距离为 60000 步，DIR 输入口为低电平时电机停止。

HG-四次谐波滤波器增益

设置或查询四次谐波滤波器增益。这个设置与四次谐波滤波器相角一起降低步进电机低速时的力矩波动。

注意：我们强烈建议您只使用 *ST Configurator* 软件来设置这个参数。

影响： 步进电机低速性能

参见： HP 指令

指令说明：

HG{参数#1}

详细说明：

指令类型	立即指令
用法	读/写
非易失性	只在使用 ST Configurator 软件设置时是非易失性的，其它的方法不是。
寄存器地址	无
参数#1	滤波器增益
-单位	整数
-取值范围	0~32767

举例

指令	驱动器返回	注释
HG8000 -		设置滤波器增益为 8000 步
HG	HG=8000	

HP-四次谐波滤波器相角

设置或查询四次谐波滤波器相角。这条指令与四次谐波滤波器增益联合使用以降低步进电机低速时的力矩波动。

我们强烈建议您使用 ST Configurator 软件设置这个参数。

影响：步进电机低速性能

参见：HG 指令

指令说明：

HP{参数#1}

详细说明：

指令类型	立即指令
用法	读/写
非易失性	只有在用 ST 指令设置时时非易失性的，其它的方法不是
寄存器地址	无
参数#1	滤波器相角
-单位	整数
-取值范围	-255~255

举例

指令	驱动器返回	注释
HP105	-	设置四次谐波滤波器相角为 105 度
HP	HP=105	

HW-手轮

设置驱动器为“手轮”模式，直到给定的数字信号或模拟量输入出现。手轮模式是一种低速跟随模式，类似与主编码器跟随模式，只是手轮是用手转动的。这条指令与 FE 指令不同，AC,DE 和 DI 指令不被使用。因为手轮的转速较低且平滑。STEP 和 DIR 口用来接入类似于编码器 A/B 信号的手轮信号。EG 指令设置跟随细分。

参见：EG 和 FE 指令；见 AT 指令里 AIN 口用于传感器输入的部分。

指令说明：

HW（参数#1）

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	输入口，输入状态
-单位	整数，字母
-取值范围	整数：3=EN 字母：L=low, H=high F=falling edge, R=rising edge

举例

指令	驱动器返回	注释
HW3L	-	运行在手轮模式，直到 EN 输入为低电平

IA-即时模拟量反馈

查询当前模拟量输入值。有两个不同的模拟量值可以访问。如果 IA 指令不带参数，将返回即时模拟量输入口的电压值与模拟量偏移值的差值，如果 IA 带参数，将返回即时模拟量值。

参见：AD,AV 和 AZ 指令

指令说明：

IA{参数#1}

详细说明：

指令类型	立即指令
用法	只读
非易失性	不是
寄存器地址	“a”(049)模拟量指令 “j”(058)模拟量输入
参数#1	模拟量输入
-单位	整数
-取值范围	无参数或 0=模拟量指令 1=模拟量输入

举例：

指令	驱动器返回	注释
IA	IA=2.5	模拟量指令值在中间
IA1	IA=4.99	模拟量输入范围接近 5 伏

例如：输入 AV1 给驱动器将模拟偏移量值设置为 1 伏。然后给 AIN (+) 和 GND (—) 端口加 4V 直流电压。输入 AI 命令后的结果将是接近 3 伏。

IC-即时电流反馈

查询当前设定的电流。

参见：CC，CI 指令

指令说明：

IC

详细说明：

指令类型	立即指令
用法	只读
非易失性	不是
寄存器地址	“c”（051）
参数#1	0.01A

举例

指令	驱动器返回	注释
IFH	-	立即指令返回值格式为十六进制
IC	IC=015E	当前设定电流为 3.5A
IC	IC=FEA2	当前设定电流为-3.5A
IFD	-	立即指令返回值格式为十进制
IC	IC=350	当前设定电流为 3.5A
IC	IC=-3.5	当前设定电流为-3.5A

ID-即时距离反馈

查询最后一个运动的开始位置到现在位置的相对距离。

指令说明：

ID

详细说明：

指令类型	立即指令
用法	只读
非易失性	不是
寄存器地址	“d”（052）
单位	步

举例

指令	驱动器返回	注释
IFH	-	立即指令返回值格式为十六进制
ID	ID=00002710	当前位置为 CW 方向 10000 步
ID	ID=FFFFD8F0	当前位置为 CCW 方向 10000 步
IFD	-	立即指令返回值格式为十进制
ID	ID=10000	当前位置为 CW 方向 10000 步
ID	ID=-10000	当前位置为 CCW 方向 10000 步

IF-立即指令格式

设置所有“I”指令的返回值数据格式为十进制或十六进制。

查询驱动器的数据有两种格式：十进制或十六进制。考虑到速度和效率，默认设置为十六进制，转换成十进制 ASCII 码格式会降低速度或轻微的延时。十六进制最小化上面的要求，以将内部的二进制数据转换成 ASCII 码格式，这样加速了发送反馈值的过程，因此给出最接近的值。特别的，在很强大的上位计算机里进行写操作，可以很容易的将十六进制码转换成十进制值。

所有“I”指令可以在任何时候，以任何频率使用，只受设定的波特率限制。立即指令在接收到后立即执行，不管驱动器的指令存储器里是不是有指令，不管什么格式，在处理响应过程中间会有一点轻微延时，所以必须小心分析数据用到的“真实时间”。

影响：立即指令 IA,IC,ID,IP,IT,IU 和 IV

指令说明：

IF{参数#1}

详细说明：

指令类型	立即指令
用法	读/写
非易失性	不是
寄存器地址	无
参数#1	返回格式
-单位	字母
-取值范围	H 或 D
-默认值	H

举例

指令	驱动器返回	注释
IFH	-	设置格式为十六进制
ID	ID=00002710	距离为 10000 步
IF	IF=H	
IF	-	设置格式为十进制
ID	ID=10000	距离为 10000 步
IF	IF=D	

IH-立即输出高电平

设置输出口立即输出高电平。如果在缓冲器指令执行完毕之前不想改变输出，可以使用 SO。

参见：IL,SO 指令

指令说明：

IH(参数#1)

详细说明：

指令类型	立即指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	输出口
-单位	整数
-取值范围	1

举例

指令	驱动器返回	注释
IH1	-	立即输出高电平

IL-立即输出低电平

设置输出口立即输出低电平。如果您在缓冲器指令执行完毕之前不想改变输出状态，可以使用 SO 命令。

参见：IH 和 SO 指令

指令说明：

IL(参数#1)

详细说明：

指令类型	立即指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	输出口
-单位	整数
-取值范围	1

举例

指令	驱动器返回	注释
IL1	-	输出口立即输出低电平

IO-输出状态

不带参数时，这条指令查询指定输出口的即时状态。以一个 8 位二进制显示，最右边一位表示输出状态。若带参数，这条指令用来设置输出口的状态，参数使用等同于二进制的十进制数，“0”表示闭合，“1”表示断开。

注意：因为 ST 只有一个输出口，所以这条指令只返回 0 或 1。

参见：IS 指令

指令说明：

IO{参数#1}

详细说明：

指令类型	立即指令
用法	读/写
非易失性	不是
寄存器地址	无
参数#1	状态
-单位	整数
-取值范围	0, 1

举例

指令	驱动器返回	注释
IO	IO=00000000	输出为低电平
IO	IO=00000001	输出为高电平
IO0	-	设置输出为低电平（闭合）
IO1	-	设置输出为高电平（断开）

IP-即时位置反馈

查询当前的绝对位置。位置数据时一个 32 位的值，如果以十六进制格式发送，它将占 8 个字符长度，如果以十进制格式发送，它的取值范围为-2147483647~2147483648

指令说明：

IP

详细说明：

指令类型	立即指令
用法	只读
非易失性	不是
寄存器地址	无
-单位	步

举例

指令	驱动器返回	注释
IFH	-	设置返回值格式为十六进制
IP	IP=00002710	绝对位置为 10000 步
IP	IP=FFFFD8F0	绝对位置为-10000 步
IFD	-	设置返回值格式为十进制
IP	IP=10000	绝对位置为 10000 步
IP	IP=-10000	绝对位置为-10000 步

IS-即时输入状态反馈

查询 STEP、DIR 和 EN 三个数字输入口的状态。

指令说明：

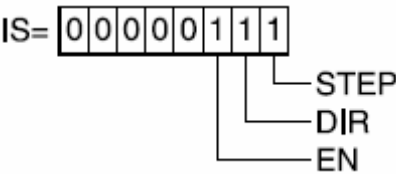
IS

详细说明：

指令类型	立即指令
用法	只读
非易失性	不是
寄存器地址	无

举例

指令	驱动器返回	注释
IS	IS=00000000	所有三个输入口都闭合
IS	IS=00000111	所有三个输入口都断开
IS	IS=00000001	STEP 输入断开，DIR 和 EN 输入都闭合
IS	IS=00000101	STEP 和 EN 输入断开，DIR 输入闭合



IT-即时温度反馈

查询驱动器的温度传感器的当前值。温度传感器的精度为 0.1 度，当温度达到一个指定的最高温度时驱动器将报警。

指令说明：

IT

详细说明：

指令类型	立即指令
用法	只读
非易失性	不是
寄存器地址	“t”（068）
单位	0.1 度

举例

指令	驱动器返回	注释
IT	IT=275	驱动器温度为 27.5℃
IT	IT=310	驱动器温度为 31.0℃

IU-即时电压反馈

查询当前输入的直流电压值，精度为 0.1 伏，当电压大于最大值或小于最小值时驱动器将报警。

指令说明：

IU

详细说明：

指令类型	立即指令
用法	只读
非易失性	不是
寄存器地址	“u”（069）
单位	0.1 VDC

举例

指令	驱动器返回	注释
如果 IF=H		
IU	IU=01E2	直流供电电压为 48.2V

如果 IF=D		
IU	IU=482	直流供电电压为 48.2V

IV-即时速度反馈

查询当前的电机速度，单位转/分

指令说明：

IV{参数#1}

详细说明：

指令类型	立即指令
用法	只读
非易失性	不是
寄存器地址	“w”（071）目标速度
参数#1	速度选择器
-单位	整数
-取值范围	1=目标速度

举例

指令	驱动器返回	注释
IV1	IV=1000	电机运行在 1000 转/分

JA-点动加速度

设置或查询点动模式的加速度/减速度，单位转/秒²。发送不带参数的 JA 指令，驱动器将返回当前的点动加速度/减速度。设置 JA，将覆盖最后一次 JA 和 JL 的值。这意味着若要设置不同的 JA 值和 JL 值，您必须先设置 JA 值来设置加速度，然后再设置 JL 来设置减速度。在点动过程中 JA 的值是不能改变的。在点动过程中使用 CS 命令来改变点动速度。

影响：CJ,WJ(jogging)指令

参见：JL，JE，JD，JS，CS，CJ，SJ

指令说明：

JA{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“K”(027)
参数#1	点动加速度值
-单位	转/秒 ²
-取值范围	0.167~5461.167
-默认值	100

举例

指令	驱动器返回	注释
JA10	-	设置加速度为 10 转/秒 ²
JA	JA=10	

JC-速度模式第二速度

设置或查询速度模式里的第二速度。用 EN 输入来选择 JC 指令设定的速度。这条指令只对 CM13,14,17 和 18 开放。

影响：速度模式

参见：AD,AG 和 CM 指令

指令说明：

JC{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	速度模式第二速度
-单位	转/秒
-取值范围	0.0042~133.0
-默认值	5

举例

指令	驱动器返回	注释
JC11	~	设置速度模式的第二速度为 11
JC	JC=11	

JD-点动禁止

在 WI 说明的时间内，禁止 STEP 和 DIR 输入作为点动输入。

影响：WI 指令执行期间的点动

参见：JE 和 WI 指令

指令说明：

JD

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无

举例

指令	驱动器返回	注释
JD	-	当激活 WI 时，禁止点动

JE-点动使能

在 WI 指令说明的期间内，使能 STEP 和 EN 输入作为点动输入。JA、JL 和 JS 指令分别定义点动加速度、点动减速度和速度。

在一次应用中，STEP 和 DIR 输入只能指派一种定义。如果您想使用 STEP 和 DIR 输入作为点动输入，可以使用 JE 来定义。如果驱动器控制模式是 CM7,11,12,13,14,15,16,17 或 18，则 JE 无效，因为这些模式预先定义了这两个输入口，且优先级高于 JE 指令。同样，在设置了 JE 指令后再设置 DL 指令为 1 或 2，将重新设定 STEP 和 DIR 输入口为限位输入，并关闭点动定义。换句话说，JE、DL 指令和控制模式 CM7,11,12,13,14,15,16,17 或 18，他们各自都给 STEP 和 DIR 定义一种用法，它们之间都互相排斥。控制模式拥有最高优先级，JE 和 DL 指令按覆盖书写的原则，后输入的有效。

影响：WI 指令期间的点动
参见：JD 和 WI 指令

指令说明：
JE
详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无

举例			
指令	驱动器返回	注释	
JE	-	WI 指令期间使能点动	

JL-点动减速度

设置或查询点动模式和速度模式的减速度，单位转/秒²。在点动期间不能改变 JL 的值。为了与老产品兼容，JA 既设置 JA，又设置 JL。如果要 JL 和 JA 不同，要先设置 JA，再设置 JL。

影响：WI 指令期间的点动，速度模式，CJ 指令

参见：JA,JE,JS,CS,CM 和 SJ 指令

指令说明：

JL{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	点动减速度
-单位	转/秒 ²
-取值范围	0.167~5461.167
-默认值	100

举例

指令	驱动器返回	注释
JL25	-	设置减速度为 25 转/秒 ²
JL	JL=25	

JS-点动速度

设置或查询点动速度，单位转/秒。

影响：WI 指令期间的点动，速度模式，CJ 指令

参见：CJ,CS,CM 和 JE 指令

指令说明：

JS{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“J”(026)
参数#1	点动速度
-单位	转/秒
-取值范围	0.0042~133.3333
-默认值	10

举例

指令	驱动器返回	注释
JS10.35	-	设置点动速度为 10.35
JS	JS=10.35	

MD-电机禁止

禁止电机接口输出，即停止给电机绕组供电。如果 BO 指令定义了输出口，禁止电机的同时也激活输出口。

影响：所有运动指令

参见：BD,BO 和 ME 指令

指令说明：

MD

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无

举例

指令	驱动器返回	注释
MD	-	停止给电机绕组供电

ME--电机使能

恢复给电机绕组供电。如果 EN 输入的状态不能使能电机时，驱动器将返回一个返回值“&”，表示不能使能驱动器。如果 BO 定义了输出口，使能电机的同时也关闭输出口功能，使能时也会有一个延时，这个延时由 BE 设置。

重点：这条指令会恢复先前的控制模式。例如，如果驱动器先前工作在速度模式，则发送 ME 指令后，电机会立即运动。如果是外接脉冲模式，要注意外接脉冲序列，以免出现意外的动作。

影响：所有运动指令

参见：BE,BO 和 MD 指令

指令说明：

ME

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无

举例

指令	驱动器返回	注释
ME	-	电机使能，恢复电机电流
ME	&	电机未使能，检查 EN 输入状态是否正确

MN-型号编号

查询驱动器型号。驱动器返回一个表示型号的单字符。

注意：与其它指令不同，其它指令的返回值时“指令=参数”的模式，而 MN 指令的返回值只有一个字符，例如“D”。

指令说明：

MN

详细说明：

指令类型	立即指令
用法	只读
非易失性	是
寄存器地址	无
单位	字符
取值范围	D=MSST5-S E=MSST10-S

举例

指令	驱动器返回	注释
MN	E	连接的驱动器时 MSST5-S

MO-运动输出

定义驱动器的数字输出口为运动输出。

ST 驱动器的数字输出有四种定义：报警输出、刹车输出、运动输出或脉冲输出。这些输出功能必须分别使用这个输出口，所以只能定义一种。有两个方法来定义输出：通过ST Configurator软件或 SCL 指令。设置输出为报警输出，使用 MO 指令和下列中的一个代码。

注意：设置AO 指令参数为1 或2，将覆盖先前对输出口的的定义，类似的，如果您在设置MO 指令为1 或2 之后使用BO 或AO 指令来定义输出口，输出口的用法将被重新定义，并自动设置AO 指令的参数为3。

MO 指令可以定义以下三种输出状态：

MO1：电机停止时输出口闭合（通电）

MO2：电机停止时输出口断开（断电）

MO3：输出口不作为运动输出使用，可以定义为其它的自动输出口或通用输出口使用。

MO4：输出口定义为脉冲输出。使用 ST 可以设置电机每转输出的脉冲数。

影响：数字输出口的定義

参见：AO 和 BO 指令

指令说明：

MO{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	状态代码
-单位	整数
-取值范围	1, 2, 3, 4
-默认值	3

举例

指令	驱动器返回	注释
MO1	-	电机停止时输出口闭合（通电）
MO	MO=1	

PB-上电波特率

设置或查询 RS-232 串行通讯上电时的波特率。当执行这条指令时，将设置波特率并立即存储到非易失性寄存器。

驱动器默认的上电波特率为 9600B,如果在 1 秒钟内没有检查到主机，且驱动器被设置为主控制器，驱动器将按照非易失性寄存器里的参数发送波特率。主机可以在任何时候改变波特率。

参见：BR,PR 和 TD 指令

指令说明：

PB{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	波特率代码
-单位	整数
-取值范围	1=9600 2=19200 3=38400 4=57600 5=115200
-默认值	1

举例

指令	驱动器返回	注释
PB2	-	设置驱动器波特率为 19200，并立即保存
PB	PB=2	

PC-上电电流

设置或查询驱动器电流。这条指令与 CC 指令不同, PC 指令将被保存到驱动器的非易失性寄存器了。

影响: 电机电流, 电机力矩

参见: CC 和 PI 指令

指令说明:

PC{参数#1}

详细说明:

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	上电电路
-单位	安培
-取值范围	0~5.0 (MSST5-S) 0~10.0 (MSST10-S)

MSST 驱动器的电流取决于电机的额定电流。

举例

指令	驱动器返回	注释
PC3.2	-	设置上电电流为 3.2A
PC	PC=3.2	

PI-上电待机电流

待机电流是电机停止时每相运行电流的一个比例值。电机电流低于运行电流有助于电机冷却。通常定义待机电流为运行电流的 50%。电机运动完最后一步后，有一个延时，延时结束后待机电流才有效。延时时间由 CD 指令设置。

PI 指令与 CI 指令不同的是，PI 指令的值被立即存储到非易失性寄存器，而 CI 指令的值则要靠 SA 指令来存储。

影响：电机停止时的电流，保持力矩

参见：CD,CI 和 PC 指令

指令说明：

PI{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	上电待机电流
-单位	0.01 安培
-取值范围	0~5.0 (MSST5-S) 0~10.0 (MSST10-S)

举例

指令	驱动器返回	注释
PI75	-	设置电机电流为 0.75 安培
PI	PI=0.75	

PM-上电模式

设置或查询驱动器的上电模式。PM 决定驱动器上电时如何通讯。PM=2 或 PM=5 表示用 SCL 通讯。

影响：串行通讯

参见：CM 指令

指令说明：

PM{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	上电模式
-单位	2-SCL/主机指令（驱动器使能） 3-ST Configurator 4-MisNet Hub 5-SCL/主机指令（驱动器禁止）

举例

指令	驱动器返回	注释
PM2	-	驱动器上电模式为 SCL 模式，或其它主机控制模式
PM	PM=2	

PR-协议

设置或查询串行通讯协议。有很多设置可以在 PR 指令里打开或关闭。每个设置在 6 位二进制字里占用 1 位，这个字用十进制表示。如果发送不带参数的 PR 指令，驱动器将返回一个相当于字的十进制数。不同的协议设置和每个位的分配见下表。

影响：RS-232 串行通讯协议

参见：BR 和 PB 指令

指令说明：

PR{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	协议代码
-单位	十进制数
-取值范围	1~63 (000001~111111) Bit0-默认值 (标准 SCL) Bit1-总是使用地址符 Bit2-Ack/Nak Bit3-校验位 Bit4-RS-485 adaptor (ST 中不用) Bit5-3 位数字代码寄存器地址
-默认值	5

举例

指令	驱动器返回	注释
PR1	-	设置标准的 SCL 协议
PR4	-	打开 Ack/Nck
PR	PR=4	

PR 指令的详细资料：

因为在主机控制模式应用中串行通讯需要的详细特征，您可以用 PR 指令调试一个最适合您应用的串行通讯协议。

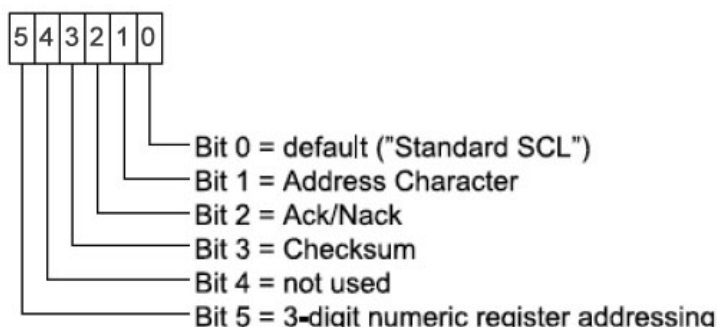
注意：我们强烈建议使用 ST Configurator 软件来设置 PR 协议，它提供相当与 PR 指令的以下选项。

在设置驱动器的时候 PR 指令就被使用过一次，并作为启动参数保存起来。尽管如此，为了改变串行通讯，可以在任何时候对它进行更改。

PR 指令的工作是靠发送的等于一个 6 位二进制的字的十进制数完成的。字的每一位表示一个不同

的设置，如下图所示，该位为“1”时，表示该项被设置，为零时表示不设置。

下面时对每一位的详细说明。



Bit0 - 默认值（标准 SCL）

PR 指令的参数不能为零，所以如果其它位都为“0”，则 Bit0 位必须为“1”。因此，这个位被认为是默认值或“标准 SCL”位。如果其它的任何一位为“1”，则 Bit 被忽略，例如，PR4 和 PR5s 是一样的。下面是当 PR=1 时一些指令的举例和响应数据包，如果指令不要求返回值，则不会发送响应，也就是说，驱动器只有在指令要求返回值时才会发送响应。

发送数据举例：

指令	驱动器返回	注释
DI8000	-	设置距离 8000 步
1DI8000	-	设置地址为“1”的驱动器的距离为 8000 步

返回数据举例：

指令	驱动器返回	注释
DI	DI=8000	返回距离
1DI	1DI=8000	返回地址为“1”的驱动器的距离

Bit1 - 地址字符（总是发送地址字符）

设置了这个选项之后，驱动器地址字符将总是与反馈数据一起装在返回值数据包里。

发送数据举例：

指令	驱动器返回	注释
VE50	-	设置速度为 50 转/秒
1VE50	-	设置地址为“1”的驱动器的速度为 50 转/秒

返回数据举例

指令	驱动器返回	注释
VE	1VE=50	设置地址为“1”的驱动器相应并且返回速度值
1VE	1VE=50	设置地址为“1”的驱动器相应并且返回 1 号驱动器的速度值

Bit2 - Ack/Nak（总是发送应答字符）

不管指令是否要求返回数据，设置这条指令后驱动器将应答主机发送的每条指令。如果指令要求返回数据（如“DI”），驱动器响应就被当作是应答；如果指令不要求返回数据，驱动器仍然会返回一个字符，如下：

“%” - 百分号是表示正确的应答字符，表示驱动器接受指令并执行。

“*” - 星号是表示有异议的应答字符，表示驱动器接受指令并存储到缓冲器的序列中，将依照缓冲器里指令序列的状态来执行。

“？” 疑问号是表示否定的应答字符，表示接收到的指令有错误，疑问号后面会跟随一个表示错误类型的字符，既错误代码，下面是错误代码：

否定应答代码

1=指令时间同步错误

2=参数长度过长

3=参数过少

4=参数过多

5=参数超出取值范围

6=指令缓冲器满

7=指令无法处理

8=程序运行中

9=密码错误

10=通讯口错误

11=字符错误

应答字符总是会输出到 RS-232 端口。如果是 2 线或 4 线的 RS-485 网络模式，应答字符将依下列情况输出。

- 1、如果指令里的起始位有地址符，将发送应答字符。
- 2、如果全局指令不要求返回数据，就不会发送应答字符。
- 3、如果指令里没有地址符，且要求返回数据，如果网络上有多个驱动器，每个驱动器都会返回数据，就会造成数据冲突。注意，在网络模式中，指令一定要带上地址，以避免冲突。

注意：在网络模式中尽量不要使用应答字符（%，*，？）作为地址符，以避免混淆。

正确指令举例：

指令	返回值	注释
DI8000	%	驱动器全部响应，设置距离为 8000，发送正确应答
1DI8000	1%	地址为“1”的驱动器响应，设置距离为 8000，发送正确应答

错误指令举例：

指令	返回值	注释
VE200	?5	驱动器全部响应，设置速度为 200，发送错误应答，因为参数 200 溢出
1VE200	1?5	地址为“1”的驱动器响应，设置速度为 200，发送错误应答，因为参数 200 溢出

寄存器指令举例：

指令	返回值	注释
AC10	*	驱动器全部响应，设置加速度为 10 转/秒 ² ，发送接受应答
1AC10	*	地址为“1”的驱动器响应，设置加速度为 10 转/秒 ² ，发送接受应答

Bit3 – 校验（8 位校验）

还没有实现，请咨询厂商。

Bit4 – ST驱动器不使用

Bit5 – 三位数字代码寄存器地址

通常情况下，访问数据寄存器使用单字母、号码或其它的 ASCII 字符。设置 Bit5 后 (Bit5=1, PR=32)，所有的数据寄存器被改为使用 3 位数字代码 (000~074) 访问 (见 数据寄存器部分)。Bit5 的设置，实现了 RL 指令和 RU 指令的使用，更实现了驱动器和 HMI 的兼容。

注意：不管 Bit5 是否设置，当驱动器返回数据时，数据寄存器总是用指定的字符表示。

RL 指令举例：

指令	返回值	注释
RL017100 -		给寄存器 017 (“A”) 赋值 100
RL017	RLA=100	驱动器返回加速度寄存器的值，寄存器用 “A” 表示

RU 指令举例

指令	返回值	注释
RU0174	RUA=100	驱动器返回从 017 寄存器开始的四个连续的寄存器地址里的内容。
	RUB=150	
	RUC=140	
	RUD=210	

PS-暂停

暂停执行缓冲器指令，直到执行 CT 指令为止。这样利于在指令缓冲器里保存一个指令序列。先给驱动器发送一个 PS 指令，暂停执行缓冲器指令，然后输入期望的指令序列，等待一个外部触发时，发送一个 CT 指令，驱动器将按顺序执行缓冲器里的指令序列。

参见：CT 指令

指令说明：

PS

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无

举例

指令	驱动器返回	注释
PS	-	暂停执行缓冲器指令

RE-重新启动

重新启动驱动器，复位错误状态，并用上电参数初始化驱动器。为了防止重新启动后电机运动，驱动器将停留在禁止状态一段时间。

参见：SA 指令里保存或改变非易失性参数的部分

指令说明：

RE

详细说明：

指令类型	立即指令
用法	只写
非易失性	不是
寄存器地址	无

RL-载入寄存器（立即指令）

设置或查询数据寄存器里的内容。数据值被当作一个长字节保存。当给一个短寄存器赋一个长字节时，只有低字节有效。

RL 指令对改变驱动器参数是非常有用的。例如，当点动正进行时，可以用修改“R”和“L”寄存器的值来改变点动加速度和减速度，改动后立即生效，而如果用 JA 指令和 JL 指令来修改点动加速度和减速度的值，则需要等待此次点动结束后，在下次点动发生时才有效。

注意：如果使用寄存器设置，参数是没有经过预处理的。这意味着用 RL 指令设置参数时，是使用驱动器内部的单位。在有些地方，驱动器内部的单位和与之相对应的指令的单位是不同的。例如：点动加速度既可以通过 JA 指令设置，也可以通过“K”寄存器设置，JA 指令的单位是转/秒²，“K”寄存器的单位是 10*转/分/秒，即 JA100 等于 RLK600，见手册后面的数据寄存器部分的关于寄存器分配和单位的详细资料。

影响：所有数据寄存器

参见：数据寄存器部分

指令说明：

RL（参数#1）{参数#2}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	不是
寄存器地址	所有数据寄存器
参数#1	数据寄存器地址
-单位	字符
-取值范围	所有读/写寄存器和用户自定义的寄存器，只读存储器可查询
参数#2	数据寄存器值
-单位	整数
-取值范围	+/-2147483647（长数据寄存器） +/-32767（短数据寄存器）

举例

指令	驱动器返回	注释
RLA150 -		设置加速度寄存器值为 1500 转/分/秒(25 转/秒 ²)
RL	RLA=150	返回加速度寄存器值
JA100 -		设置加速度为 100 转/秒 ²
JS1 -		设置点动速度为 1 转/秒
CJ -		开始点动
JA10 -		设置加速度为 10 转/秒 ²

CS5	-	改变点动速度为 5 转/秒
RLK60	-	设置加速度为 10 转/秒 ²
CS1	-	改变速度到 1 转/秒
SJ	-	停止点动

如果 PR 指令 Bit5 被设置

RL017100	-	设置加速度寄存器为 1000 转/分/秒
RL017	RL017=100	查询加速度寄存器值

RS-查询状态

查询驱动器当前状态。也就是查看驱动器当前在做什么。驱动器不同项目的状态都用一个对应的字符表示，一次可以返回多个字符表示当前所处状态。

参见：SC 指令

指令说明：

RS

详细说明：

指令类型	立即指令
用法	只读
非易失性	不是
寄存器地址	无
单位	字符代码
取值范围	A=报警出现 D=驱动器禁止 E=驱动器出错 H=正在找原点 J=正在点动 F=正在运动（Feed&Jog 指令） M=正在运动（任意输入信号） R=准备完毕

举例

指令	驱动器返回	注释
RS	RS=R	驱动器使能且准备完毕
RS	RS=ADE	报警出现，驱动器出错并且停止
RS	RS=JR	电机点动，驱动器使能

RU-寄存器上传

上传一段数据寄存器的内容。RU 指令一次可上传多达 16 个寄存器的值，每返回一个寄存器值都以一个回车键结束

影响：所有数据寄存器

参见：RL 和 PR 指令

指令说明：

RU (参数#1) (参数#2)

详细说明：

指令类型	立即指令
用法	只读
非易失性	是
寄存器地址	所有寄存器
参数#1	寄存器地址
-单位	字符
-取值范围	所有寄存器
参数#2	上传连续数据寄存器数目
-单位	整数
-取值范围	1-16

举例

指令	驱动器返回	注释
RUA5	RUA=400	加速度值
	RUB=400	减速度值
	RUC=16000	改变距离值
	RUD=8000	距离值
	RUE=0	编码器值

若 PR 指令设置了 Bit5

RU0175	RUA=400	加速度值
	RUB=400	减速度值
	RUC=16000	改变距离值
	RUD=8000	距离值
	RUE=0	编码器值

注意：所有数据寄存器使用驱动器内部的单位，内部的单位和指令的单位是不同的。

RV-修正版本

查询驱动器硬件版本。版本用一个没有小数点的三位十进制数表示。见下例。

指令说明：

BE{参数#1}

详细说明：

指令类型	立即指令
用法	只读
非易失性	是
寄存器地址	无
单位	驱动器硬件版本

举例

指令	驱动器返回	注释
RV	RV=150	驱动器硬件版本是 1.50
RV	RV=222	驱动器硬件版本是 2.22

SA-保存参数

保存指令的参数到非易失性寄存器。发送 SA 指令后，所有非易失性指令的参数都被保存。这条指令可修改驱动器的上电默认值。

参见：RE 指令

指令说明：

SA

详细说明：

指令类型	立即指令
用法	只写
非易失性	不是
寄存器地址	无

举例

指令	驱动器返回	注释
SA	-	保存非易失性指令参数

SC-状态代码

这条指令类似于 RS 指令，只是 SC 指令返回的驱动器状态用十六进制表示，其中的每一位都表示一个状态。这样，主机可以快速的一次查询驱动器的多个状态。

参见：RS 指令

指令说明：

SC

详细说明：

指令类型	立即指令
用法	只读
非易失性	是
寄存器地址	无
单位	十六进制码的二进制状态代码

举例

指令	驱动器返回	注释
SC	SC=0009	驱动器使能且在运动
SC	SC=0004	驱动器出错且禁止
SC	SC=0209	驱动器报警，正在运动，且使能

状态代码位分配：

十六进制值	位	说明
0001	0	驱动器打开/使能
0002	1	不使用
0004	2	驱动器出错
0008	3	正在运动（外脉冲）
0010	4	正在运动（Feed 运动）
0020	5	点动
0040	6	停止（ST/SK）
0080	7	等待（WI）
0100	8	保存（SA）
0200	9	报警（AL）
0400	10	正在找原点（SH）
0800	11	等待延时（WT）
1000	12	不使用
2000	13	不使用
4000	14	不使用
8000	15	初始化

SF-步进滤波器频率

设置或查询步进滤波器频率。主要的应用是把微分加入到步进电机的运动中，当驱动器设置为低细分时，SF 指令使电机运动平滑。当使用一个低能力的分度器或编码器给驱动输入脉冲并且需要电机平滑运动时 SF 指令尤其有效。

指令说明：

SF{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	步进滤波器频率
-单位	赫兹
-取值范围	0~16000
-默认值	2500

举例

指令	驱动器返回	注释
SF500	-	设置步进滤波器频率为 500 赫兹
SF	SF=500	

SH-找原点

执行 SH 指令。需要指定原点传感器的输入口和输入状态。VE、AC 和 DE 分别设置速度、加速度和减速度，DI 指令设置方向和距离（“-”为 CCW，否则为 CW）。

SH 指令组合了多个动作。基本动作类似于一个 FS 和 FP 的指令组合，电机先运动到原点传感器，并记下这个绝对位置，然后开始减速，减速后在运动到这个绝对位置。如果电机运动到限位时还没有找到原点，则向相反方向运动，直到遇到另一个限位，然后重新找原点，这样确保找到原点时保持原来的方向。

注意：这条指令需要三个传感器输入到三个数字输入口：一个原点传感器，一个 CW 限位和一个 CCW 限位。因此，只能将原点传感器接到 EN 输入口。

参照：DL 指令

指令说明：

SH{参数#1}

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	输入口，输入状态
-单位	整数，字母
-取值范围	整数：1=STEP,2=DIR,3=EN 字母：L=low, H=high F=falli edge, R=rising edge

举例

指令	驱动器返回	注释
SH3L	-	原点传感器从 EN 输入口输入，低电平有效。

SI-使能输入

定义 EN 输入为使能输入。

在一次应用中，EN 输入只能指定一种定义，如果您想将 EN 输入用作使能输入，有以下两种方法可以定义：用 ST Configurator 软件或 SI 指令。如果控制模式为 CM13、14、17 和 18，SI 将不起作用，因为这些模式将 EN 输入作为速度改变输入，且优先级高于 S 指令。同样，在设置 SI 指令后设置 AI 指令，会将 EN 输入重新指定为驱动器报警复位输入，并自动关闭任何使能用法（SI3）。换句话说，SI 指令、AI 指令和控制模式 CM13、13、14、17 和 18，各自给 EN 输入定义一种用法。它们各自排斥别的指令使用 EN 输入，控制模式有最高优先级，且总是不断的防止 AI 指令和 SI 指令使用 EN 输入，AI 指令和 SI 指令用覆盖写入的方式排斥另一个使用 EN 输入。

SI 指令可以定义以下三种使能输入状态：

AI1：当 EN 输入端闭合（通电）时驱动器使能。

AI2：当 EN 输入端断开（断电）时驱动器使能。

AI3：EN 输入不用于使能输入，可以作为一个通用输入口来使用。如果设置 SI 后，CM 设置为 13、14、17 和 18，或 AI 设置为 1 或 2，则 SI 被自动设置为 3。

参照：AI 和 CM 指令

指令说明：

SI{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	使能输入状态
-单位	整数
-取值范围	1, 2, 3
-默认值	3

举例

指令	驱动器返回	注释
SI1	-	EN 输入口断开时，驱动器使能
SI	SI=1	

SJ-停止点动

停止点动运动。

影响：CJ 指令

参见：JA,CJ,ST,SK 和 SM 指令

指令说明：

SJ{参数#1}

详细说明：

指令类型	立即指令
用法	只写
非易失性	不是
寄存器地址	无

举例

指令	驱动器返回	注释
SJ	-	停止点动

SK-停止并清除缓冲器

停止所有缓冲器指令的进程，并清除指令缓冲器里的指令。此时的减速度由 AM 指令设定。如果使用了“D”作参数，减速度将有 DE 和 JA 定义。

影响：运动和指令缓冲器内容

参见：AM 和 ST 指令

指令说明：

SK{参数#1}

详细说明：

指令类型	立即指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	减速度
-单位	字母
-取值范围	D=减速度由 DE 或 JA 指令设置 无参数=减速度由 AM 指令设置

举例

指令	驱动器返回	注释
SK	-	停止运动并清除指令缓冲器，AM 设置减速度
SKD	-	停止运动并清除指令缓冲器，DE 或 JA 指令设置减速度

SO-设置输出口

设置驱动器的数字输出为给定的状态，高电平或低电平。这条指令只有在数字输出口没有被定义为专用输出口-AO、BO、MO-时才能使用。

参见：IL,IH,IO 指令

指令说明：

SO{参数#1}

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	输出口，输出状态
-单位	整数，字母
-取值范围	整数：1=数字输出口 字母：L=low，H=high

举例

指令	驱动器返回	注释
SO1L	-	1 号数字输出口状态为低电平
SI1H	-	1 号数字输出口状态为高电平

SP-设置位置

设置或查询电机当前的绝对位置，单位：步。

参见：FP 指令

指令说明：

SP{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	不是
寄存器地址	无
参数#1	绝对位置
-单位	步
-取值范围	+/-2, 147, 483, 647

举例

指令	驱动器返回	注释
SP100	-	设置绝对位置为 100 步
SP	SP=100	

SS-发送串

指示驱动器返回指定的字符串（最多 4 个字符）。这条指令可以使得主机知道指令序列已执行完毕。指令缓冲器可以同时存储几条 SS 指令，

指令说明：

SS（参数#1）

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	字符串
-单位	任何显示的字符
-取值范围	最多 4 个字符

举例

指令	驱动器返回	注释
AC100	-	设置加速度为 100 转/秒 ²
DE100	-	设置减速度为 100 转/秒 ²
VE1	-	设置速度为 1 转/秒
EG10000	-	设置细分为 10000 步/转
DI100000	-	设置距离为 100000 步
FL	-	运动到指定距离
Ssdone	-	FL 执行完毕时，发送字符串“done”到主机

ST-停止

停止当前执行的缓冲器指令，但是不影响其它指令缓冲器里缓冲器指令。ST 指令的减速度由 AM 指令设定，如果 ST 指令用了“D”作参数，则减速度由 DE 或 JA 指令设定。

参见：SK 指令

指令说明：

ST{参数#1}

详细说明：

指令类型	立即指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	减速度
-单位	字母
-取值范围	D=减速度由 DE 或 JA 指令设置 无参数=减速度由 AM 指令设置

举例

指令	驱动器返回	注释
ST	-	停止运动，减速度由 AM 设置
STD	-	停止运动，减速度由 DE 或 JA 设置

TD-传输延时

设置或查询驱动器响应指令后返回一个返回值的延时。

影响：RS-232 串行通讯

参见：BR,PB 和 PR 指令

指令说明：

TD{参数#1}

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	无
参数#1	延时时间
-单位	毫秒
-取值范围	0~32767
-默认值	0

举例

指令	驱动器返回	注释
TD10	-	设置延时为 10 毫秒
TD	TD=10	

VC-改变速度

设置或查询改变速度。

影响：FC,FD 指令

指令说明：

VC（参数#1）

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“V”（037）
参数#1	改变速度值
-单位	转/秒
-取值范围	0.0042~133.3333
-默认值	5

举例

指令	驱动器返回	注释
VC5	-	设置改变速度为 5 转/秒
VC	VC=5	
AC100	-	设置加速度为 100 转/秒 ²
DE100	-	设置减速度为 100 转/秒 ²
DI100000	-	设置距离为 100000 步
DC75000	-	设置改变速度为 75000
VE5	-	设置速度为 5 转/秒
VC1	-	设置改变速度为 1 转/秒 ²
FC	-	执行 FC 指令

VE-速度

设置或查询“Feed”运动的速度。

影响：所有“F”指令

指令说明：

VE（参数#1）

详细说明：

指令类型	缓冲器指令
用法	读/写
非易失性	是
寄存器地址	“V”（038）
参数#1	运动速度
-单位	转/秒
-取值范围	0.0042~133.3333
-默认值	10

举例

指令	驱动器返回	注释
VE2.5	-	设置速度为 2.5 转/秒
VE	VE=2.5	
DI-20000-		设置距离为 20000，方向 CCW
FL	-	执行 FL 指令

WI-等待输入

等待输入口的指定输入状态。在 CM21、22 模式下，JE 指令激活后，在 WI 指令执行期间 STEP 和 DIR 输入口作为点动输入。

影响：STEP 和 DIR 输入口用作点动输入

参见：FI,JE 和 JD 指令

指令说明：

WI{参数#1}

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	输入口，输入状态
-单位	整数，字母
-取值范围	整数：1=STEP,2=DIR,3=EN 字母：L=low, H=high F=falli edge, R=rising edge

举例

指令	驱动器返回	注释
WI3R	-	等待 EN 输入上升沿时有效

WT-等待时间

设置一个延时，单位：秒。分辨率为 0.01 秒。

指令说明：

WT（参数#1）

详细说明：

指令类型	缓冲器指令
用法	只写
非易失性	不是
寄存器地址	无
参数#1	时间
-单位	秒
-取值范围	0.00~320（分辨率 0.01 秒）

举例

指令	驱动器返回	注释
WT2.5	-	设置延时时间为 2.5 秒

数字寄存器

指令清单里，有许多指令的作用是将数据发送到驱动器，以备后用。传送的数据就存储在数据寄存器里，直到被新的数据覆盖，或是驱动器掉电。例如，发送“VE10”，数据“10”就会保存在速度数据寄存器里，在这之后使用 FL、FP、FS 指令时，数据“10”就是最大速度值，除非重新设置最大速度或掉电才会发生改变。

不是所有的指令都会发送数据到数据寄存器，也不是所有数据寄存器都有对应的指令。如果要访问一个没有指令的数据寄存器，可以通过使用 RL 或 RX 指令，和寄存器分配的特定的标识符（见下页的数据寄存器的分配，包括数据寄存器列表和标识符）来实现。RL 指令保存数据到寄存器，RU 从寄存器读取数据。

注意：在使用 RL 和 RU 指令时，使用的是驱动器内部单位，与指令单位是不同的。

在 ST 里，有两种不同的寄存器：只读寄存器和读/写寄存器。

只读寄存器

只读寄存器用于存放驱动器的参数、设置和状态等信息，包括了控制电流、模拟量输入量、驱动器温度和内部总线电压的寄存器，等等。只读寄存器用小写字母表示。

读/写寄存器

读/写寄存器用于存放用户设置的驱动器参数和运动参数，包括加速度、速度、距离、设置电流，等等。大多数读/写寄存器都分配了一个相关联的指令，所以既可以通过 RL、RU 指令来访问，也可以通过关联指令来访问。读/写寄存器用大写字母表示。

数据寄存器分配

下面是 ST 驱动器的全部数据寄存器清单，在表格中，“标识符”表示寄存器分配的特定的标识符，“描述”说明数据寄存器的名称，“地址”表示由三个数字构成的寄存器地址，“数据类型”说明此数据寄存器是 16 位（短）还是 32 位（长）寄存器，“单位”说明驱动器如何使用该参数，“默认值”说明生产厂对此寄存器的默认值。

只读寄存器：a~z

标识符	描述	地址	数据类型	单位	默认值
a	模拟量指令值	049	短	32767=+10, -32767=-10	
c	控制电流	051	短	0.01A	
d	关联距离	052	长	步	
f	报警代码	054	长	见 AL 指令	
g	传感器位置	054	短	步	
i	驱动器输入状态	057	短	十进制表示的二进制数	
ST 驱动器的 i 寄存器里装了一个等于 16 位二进制数的十进制数，表示驱动器的数字输入输出状态，因为只有 3 个数字输入和 1 个数字输入，所以只用到 1, 2, 3, 8 位，其他的位不用，都为 0。 数字输入，“0”表示闭合，数字输出“1”表示闭合。					

j	模拟量输入 1	058	短	ADC	
i	绝对位置	060	短	Bit pattern	
m	控制模式	061	短	模式代码	
n	速度模式状态	062	短	状态代码（见下表）	
RLn<cr>指令的响应:					
等待中	0	在速度模式中等待指令			
运动中	1	速度模式运动进行中（jogging）			
快速停止	2	停止速度模式运动（无参数的 SK 或 ST 指令）			
停止中	3	停止速度模式运动（SJ、SKD 或 STD）			
结束	4	结束最后一步（1 个脉宽调制周期，62 微秒）			
o	模拟量指令值	063	短	状态代码（见下表）	
Rlo<cr>指令响应:					
描述	十进制值	注释			
等待中	0	位置模式，等待指令			
等待刹车	1	等待刹车释放			
计算中	2	计算，为动作做准备			
加速中	3	加速到最大速度			
改变速度	4	正改变速度			
速度固定	5	运动在设定的速度			
减速中	6	减速到停止			
快速减速中	7	正在快速减速			
定位中	8	正在完成最后一步			
s	状态代码	067	短	见 AL 指令	
t	驱动器温度	068	短	0.1℃	
u	Bus voltage	069	短	0.1V	
w	目标速度	071	短	0.25 转/分	

读/写寄存器地址：A~Z

大多数读/写寄存器都分配了特定的指令。在下表中，关联的指令显示在“描述”栏的括号内。

标识符	描述	地址	数据类型	单位	默认值
A	加速度（AC）	017	短	10 转/分/秒	
B	减速度（DE）	018	短	10 转/分/秒	
C	改变距离（DC）	019	长	步	
D	距离（DI）	020	长	步	
F	其它标记	022	长	二进制	
RLF<cr>指令响应:					
描述	十六进制	十进制			
距离限制标记	0x0001	1			
传感器发现标识	0x0002	2			
硬件过流	0x0004	4			
软件过流	0x0008	8			
错误的电流偏移一相 A	0x0010	16			
错误的电流偏移一相 B	0x0020	32			

绕组开路 — 相 A 0x0040 64					
绕组开路 — 相 B 0x0080 128					
说明：要清除标识，发送 RLF0<cr>					
H	保留位	024	长		
J	点动速度（JS）	026	短	0.25 转/分	
K	点动加速度（JA）	027	短	10 转/分/秒	
L	点动减速度(DE)	028	短	10 转/分/秒	
N	连续电流/改变电流	030	短	0.01A	
O	峰值电流/待机电流	031	短	0.01A	
P	绝对位置指令	032	长	步	
Q	保留位	033	长		
R	细分（EG）	034	短	步/转	
S	确定的步进位置	035	长	步	
U	改变速度（VC）	037	短	0.25 转/分	
V	速度（VE）	038	短	0.25 转/分	
X	模拟量位置增益（AP）	040	短	ADC 比例	
Y	模拟量死区（AT）	041	短	VDC 数量	
Z	模拟量偏移（AO）	042	短	0.001V	

附录 A：主机串行通讯

当 ST 驱动器处于主机控制模式时，主机需要通过串口连接发送指令到驱动器。典型的主机设备有：

- Windows-based PC 机——安装 MOONS' 软件
- 工业 PC 机——运行专业的应用软件
- PLC——有发送文本流的 ASCII 模块或串口通讯
- HMI——有发送文本流的串口通讯

ST 驱动器在主机模式下需要的配置：

- 主机串行通讯的普通结构
- 硬件——连接主机与驱动器的配线
- COM 口设置——UART 设置和波特率设置
- 通讯协议
- 通讯的详细资料
- 通讯错误

主机串行通讯的普通结构

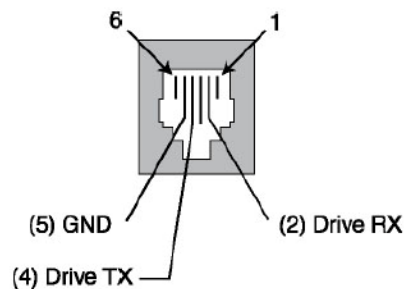
ST 的主机通讯协议是基于通用的 ASCII 码设置，在 RS-232 硬件设置用标准 UART 传送信息。

使用 ASCII 码是因为其通用性和容易理解，且方便阅读。UART 串行收发器广泛应用于各种设备，包括大多数 PC 机，且提供了一个通用格式的串行通讯界面。RS-232 的硬件连接广泛用于 UART，且提供了简单的和最通用的连接格式。

硬件

每个 ST 驱动器配备了一根 RS-232 通讯电缆。通过这根电缆，使用 ST Configurator 软件配置驱动器，和使用 SCL Utility 软件发送 SCL 指令。例如，如果您的主机是 PC 机，您就可以使用这根电缆。

如果主机是 PLC 或 HMI，您可能要自己制作通讯电缆，如图中所示，是 RS-232 通讯接口的管脚分布，驱动器的接口是一个 RJ-11 接口，Pin5 接信号地，Pin4 是 TX 脚，Pin2 是 RX 脚。



COM 口设置

UART 设置：1 个开始位，8 位数据位，无校验位，1 个停止位。

波特率设置：所有 MOONS' 的驱动器的波特率都默认为 9600bps，在大多数案例中，对于设备、设置、程序以及主机通讯，这个速度已足够。如果需要，可以设置更高的速度。在所有应用中，驱动器已波特率 9600bps 启动，如果“上电签名”被主机识别，就保留这个参数，如果上电周期结束了主机还没有收到上电签名，驱动器将激活新的波特率。

设置波特主要看您的应用，如果一个主机设备通过一个网络控制驱动器，就需要一个高速的波特率来保障通讯。

通讯协议

通常，主机与驱动器之间的通讯协议相当简单。ST 驱动器不主动发起通讯，所以驱动器通常处于接收状态。一个通讯信息包，包括所有必须的指令或响应的通讯信息的字符。也就是说，主机发起的通讯是发送一个指令信息包，而驱动器的响应是发送一个响应数据包。

指令传送（主机→驱动器）：

指令传送要求在限定的时间内以一个信息包的格式发送所有要求的字符。从接收到数据包开始，驱动器开始计算字符间的间隙时间，每收到一个字符，内部定时器被复位到 200 毫秒，如果在接收到下一个字符之前，定时器已经变为零了，驱动器将终止信息包的分析，并根据协议判断是否发送错误响应信息包。时间间隙的一部分是当有错误传送发生时让驱动器自动清除缓冲器。

响应传送（驱动器→主机）：

收到主机的指令信息包后，驱动器通常会发送一个响应信息包。响应信息包的每两个字符间都有一个时间间隙，当波特率位 9600bps 时，字符间的间隙小于一个位空间（0.0001s）。

协议设置（PR 指令）：

PR 指令用于给整体的通讯协议添加项目，也就是建立指令和响应信息包的结构。通常，当主机发送一个指令信息包给驱动器，驱动器可能识别，也可能不能识别，如果驱动器识别指令，就会执行指令，如果驱动器不能识别指令，就不能执行指令。大多数案例中，主机想知道驱动器是否识别了指令，这样，可以将驱动器设置成自动发送一个正确应答或错误应答响应信息包。

通讯详细说明

传输延时（TD 指令）：

驱动器处理完指令信息包后，有一个延时，然后才会发送响应数据包，这个延时就是传输延时，由 TD 指令设定。

注意：传输延时是为两线传输的 RS-485 串行通讯设置的，因为 ST 驱动器只能使用 RS-232 串行通讯，所以在这里没必要设置传输延时。

通讯信息包

通讯信息包的内容包括指令或响应所必须的全部字符，这些内容可以通过 PR 指令设定。

驱动器启动：

在驱动器上电时，所有驱动器都会发送“上电签名”，通知主机驱动器已上电，然后等待主机的指令。这是唯一由驱动器发起的通讯。在 MOONS' 的软件中，“上电签名”是必须的过程，否则可能会导致驱动器损坏。“上电签名”与其它指令不同，它是二进制码字符，而不是十六进制码。“上电签名”信息包包括三个字符，第一个是“225”，它告诉驱动器这个信息包是“上电签名”，第二个是硬件版本，第三个是驱动器型号。

“上电签名”是以固定波特率 9600bps 发送的，而不是按 BR 或 PB 指令的设置。如果使用 MOONS'

软件，它将响应驱动器的“上电签名”，并继续以 9600bps 进行通讯，如果不使用 MOONS' 软件，“上电签名”将终止联接，并以由 PB 或 BR 指令设置的波特率进行通讯，或以其它方式。

通讯错误

在主机向驱动器发送信息的时候，有两种错误可能出现。

硬件错误：

硬件错误会在 ST 驱动器的 LED 灯上显示出来（见附录 B 的 LED 显示代码），而驱动器不会自动发送响应，因此，主机必须通过 AL、RS 或 SC 指令来检查硬件错误。一旦确定是通讯错误，可以通过 CE 指令检查错误类型。

解析错误：

解析错误是指当驱动器接收到一条指令而不能解析。解析错误可以自动发送错误响应，由 PR 指令设定。

附录 B：LED 显示代码

ST 驱动器由两个显示错误的 LED，一个红色，一个绿色，它们用不同的闪烁次数组合表示错误类型

		描述
1	1	不使用
1	2	没有运动（驱动器禁止是尝试运动）
2	1	CCW（反转）限位
2	2	CW（正转）限位
3	1	过温
3	2	不使用
4	1	供电电源过压
4	2	供电电源欠压
5	1	过流/短路
5	2	电机不匹配（电阻超出范围）
6	1	电机绕组开路
6	2	不使用
7	1	通讯错误
7	2	不使用



上海安浦鸣志自动化设备有限公司
中国 上海 闵北工业区 鸣嘉路 168 号

电话: 021-52634688

传真: 021-62968682

邮箱: info@moons.com.cn

网址: www.moons.com.cn

ST 驱动器的 SCL 使用手册