

# SIEMENS西门子 S-1FL2中惯量型电机 1FL22032AG101HC0

产品名称	SIEMENS西门子 S-1FL2中惯量型电机 1FL22032AG101HC0
公司名称	浔之漫智控技术(上海)有限公司
价格	.00/件
规格参数	西门子:原装正品 驱动器电机电缆:假一罚十 德国:现货包邮
公司地址	上海市松江区石湖荡镇塔汇路755弄29号1幢一层 A区213室
联系电话	15801815554 15801815554

## 产品详情

**功能概述 (S7-1500) 传输顺序** 此类指令为最多 16 个变频器从站循环处理数据传输。一次只能为一个变频器激活一个作业。性能特性：  
根据总线组态为通信创建数据存储区 执行和监视 PKW 作业 监视整个系统和故障排除 与 CPU 进行通信 访问变频器功能 读取变频器参数 写入变频器参数 编程 - 使用指令进行通信 (S7-1500) 点对点编程概述 (S7-1500) 使用自由口或 3964(R) 通信进行数据交换 必须在相应 CPU 用户程序的数据块或位存储器地址区中提供发送数据。接收数据的通信模块中提供接收缓冲区。在数据块中设置相应数据块。在 CPU 的用户程序中，以下指令用于执行 CPU 与通信模块之间的数据传输。 Send\_P2P Receive\_P2P 接收缓冲区可以通过 Receive\_Reset 指令删除。通过用户程序的动态组态 作为通信模块的组态/参数分配 (页 2072)部分中所述的通信模块接口参数分配的替代或补充，建议在特定应用领域动态建立通信，即通过程序控制具体应用的通信。数据传输步骤  
该主站可确保在帧中进行循环数据传输。该主站使用作业帧对所有从站设备逐个进行寻址。被寻址的节点通过一个响应帧进行响应。接收作业帧之后，从站必须按照主站-从站过程向主站发送响应帧。只有这样，主站才能对下一个从站进行寻址。帧中的数据域  
数据域分为两个区域：参数区 (PKW) 和过程数据区 (PZD)。 STX LGE ADR 参数 (PKW) 过程数据 (PZD) BCC 参数区 (PKW) PKW 区域处理两个通信伙伴（例如控制器和变频器）之间的参数传输。例如，这包括读取和写入参数值以及读取参数说明和关联的文本。PKW 接口通常包含操作和显示、维护 和诊断作业。过程数据区 (PZD) PZD 区包含自动化操作所需的信号： - 控制字和设定值（从主站到从站） - 状态字和实际值（从从站到主站） 参数区和过程数据区的内容由从站变频器进行定义。有关这方面的其它信息，请参考变频器文档。

也可以在运行期间通过以下其中一个“Config”指令更改在通信模块的属性对话框中分配的所有参数。

Port\_Config、Send\_Config、Receive\_Config、P3964\_Config 点对点通信的程序调用 - 顺序

下图显示了用于用户程序与通信伙伴之间通信的点对点指令的功能。PtP 指令应用指令说明在

CPU、通信模块和通信伙伴之间进行数据交换（通信）。Send\_P2P 指令

Send\_P2P（发送点对点数据）可用于向通信伙伴发送数据。调用指令 Send\_P2P

以通过自由口协议发送数据。在指令的输出参数中接收到相应确认前，您必须循环调用该指令。

注意：在 XON/XOFF 数据流控制的参数分配期间，用户数据不可包含任何已组态的 XON 或 XOFF

字符。默认设置为 DC1 = 11H (XON) 和 DC3 = 13H (XOFF)。Receive\_P2P 指令

Receive\_P2P（接收点对点数据）可用于从通信伙伴中获取通信模块中接收的消息。循环调用

Receive\_P2P 指令以通过自由口协议接收数据。如果新接收的数据可用，指令将在 NDR

参数中加以指示。为了表示消息传输的开始和结束，需要在识别消息的开始和结束的自

由口协议中定义标准。检测接收缓冲区 Receive\_Reset 指令

Receive\_Reset（删除接收缓冲区）允许清除通信模块的接收缓冲区。接口或端口的动态参

数分配（可选）Port\_Config 您可以使用 Port\_Config 指令（端口组态）来组态基本接口参数，如

数据传输率、奇偶校验和数据流控制（通过用户程序动态执行）。Send\_Config 通过指令

Send\_Config（发送组态），您可以为点对点通信接口动态地组态串行发送参数，如 RTS ON 延迟/RTS

OFF 延迟。Receive\_Config 指令 Receive\_Config（接收参数分配）允许将串行接收参数动态分配

给通信模块。该指令可组态指定所接收消息的开始和结束的条件。P3964\_Config

指令 P3964\_Config（组态协议）可用于动态组态程序 3964(R) 的协议

参数，例如字符延迟时间、优先级和块检查（使用程序）。操作 RS232 伴随信号 Signal\_Get 通过

Signal\_Get 指令（获取 RS232 信号），您可以读取 RS232 信号的当前状态。Signal\_Set 通过 Signal\_Set

指令（获取 RS232 信号），可以设置 RS232 信号 DTR 和 RTS 的状态。启用 Modbus CRC 支持

和诊断中断 Get\_Features 可使用 Get\_Features 指令（获取扩展功能）获取有关 Modbus 支持和

有关生成诊断报警的信息。Set\_Features 如果模块支持，可使用指令

Set\_Features（设置扩展功能）激活诊断报警的生成。自由口或 3964(R) 通信的设置步骤

要求：通信模块的设备视图以及属性对话框中 CPU 和通信模块的组态和参数分配均已完成。1. 在 CPU

的项目导航中，选择文件夹“程序块”（Program blocks），然后双击打开文件夹中的 Main

(OB1)。程序编辑器随即打开。2. 从“指令”（Instructions）任务卡的“通信”（Communication）

区域中选择指令 Send\_P2P 和 Receive\_P2P 并将它们拖放到 Main (OB1) 的网络中。3. 按照规范组态指令。

4. 将硬件组态和用户程序下载到 CPU 中。Modbus 编程概述 (S7-1500) Modbus 通信的程序调用 - 顺序

下图所示是用户程序和 Modbus 设备之间通信的 Modbus 指令的功能。（下游使用

Send\_P2P、Receive\_P2P 和 Config 指令）。Modbus 指令应用指令说明在用户程序和 Modbus 设备之间

进行数据交换（通信）Modbus\_Master Modbus\_Master 指令允许通过 PtP 端口作为 Modbus

主站进行通信。利用 Modbus\_Master 指令，CPU 可用作 Modbus RTU 主站设备，与一个或多个 Modbus

从站设备进行通信。Modbus\_Slave Modbus\_Slave 指令允许通过 PtP 端口作为 Modbus 从站进行通信。

利用 the Modbus\_Slave 指令，CPU 可用作 Modbus RTU 从站设备，与一个 Modbus 主站设备进行通信。

接口和协议的参数分配（可选）Modbus\_Comm\_Load 指令 Modbus\_Comm\_Load 允许组态 Modbus RTU

的通信模块端口。必须运行 Modbus\_Comm\_Load 来设置 PtP 端口参数，例如：数据传

输率、奇偶校验和流控制。为 Modbus RTU 协议组态完接口后，它只能由 Modbus\_Master 或

Modbus\_Slave 指令使用。说明交替使用 Modbus\_Slave 和 Modbus\_Master

通信模块既可充当主站也可充当从站。设置 Modbus 通信的步骤

要求：通信模块的设备视图以及属性对话框中 CPU 和通信模块的组态和参数分配均已完成。1. 在 CPU

的项目导航中，选择文件夹“程序块”（Program blocks），然后双击打开文件夹中的 Main

(OB1)。程序编辑器随即打开。2. 根据您的任务，从“指令”（Instructions）

任务卡的“通信”（Communication）区域中为 Modbus 通信选择相应指令并将它们拖放到 Main (OB1)

的网络中：- 指令 Modbus\_Comm\_Load 可为 Modbus 通信组态通信模块的端口。报告 DONE（或

ERROR）前，必须在 Main (OB1) 中调用 Modbus\_Comm\_Load。- Modbus\_Master 指令用于 Modbus

主站功能。- Modbus\_Slave 指令用于 Modbus 从站功能。3. 按照规范组态指令。4.

将硬件组态和用户程序下载到 CPU 中。

变频器之间通信的 USS 指令的功能。（下游需要使用指令 Send\_P2P、Receive\_P2P 和 Config 指令）。USS 指令应用指令说明在 CPU、通信模块和 USS 变频器之间进行通信。USS\_Port\_Scan USS\_Port\_Scan 指令允许在 USS 网络中通过通信模块与最多 16 个变频器进行通信（必须循环调用）。USS\_Port\_Scan 指令通过 PtP 通信端口控制 CPU 和变频器之间的通信。每次调用此功能时，将进行与变频器之间的通信。需要执行一次指令 USS\_Port\_Scan：

由于大多数变频器都具有根据超时来监视通信完整性的可组态内部功能，因此应从时间控制的 OB 中调用 USS\_Port\_Scan 指令。与 USS 变频器进行数据交换 USS\_Drive\_Control USS\_Drive\_Control 指令允许为变频器准备发送数据并显示接收数据。

指令的输入和输出与变频器的状态和操作功能相对应。每个变频器必须调用一次 USS\_Drive\_Control 指令。对于 USS 网络，只需将一个通用背景数据块用于指令 USS\_Drive\_Control 的全部调用。针对 USS 网络，将指令 USS\_Drive\_Control 的所有调用与同一背景数据块互联。应从主程序的循环 Main (OB1) 中调用 USS\_Drive\_Control 指令。读取或修改 USS 驱动器中的参数 USS\_Read\_Param USS\_Read\_Param 指令允许从变频器中读取参数。使用 USS\_Read\_Param 指令读取控制变频器内部功能的变频器操作参数。应从主程序的循环 Main (OB1) 中调用 USS\_Read\_Param 指令。USS\_Write\_Param USS\_Write\_Param 指令允许更改变频器中的参数。应从主程序的循环 Main (OB1) 中调用 USS\_Write\_Param 指令。设置 USS 通信的步骤要求：通信模块的设备视图以及属性对话框中 CPU 和通信模块的组态和参数分配均已完成。

1. 在 CPU 的项目树中，选择“程序块” (Program blocks) 文件夹，然后双击打开所需的时间控制的 OB。程序编辑器随即打开。
2. 从“指令” (Instructions) 任务卡的“通信” (Communication) 区域中选择指令 USS\_Port\_Scan 并将其拖放到时间控制 OB 的网络中。USS\_Port\_Scan 指令允许通过 USS 网络通信。
3. 在 CPU 的项目导航中，选择文件夹“程序块” (Program blocks)，然后双击打开文件夹中的 Main (OB1)。程序编辑器随即打开。