ER 系列工业机器人 ModbusTCP 调试手册

(**RCS2** V2.0)

修订记录

序号	日期	版本	描述
1	2018.08.17	V1.0	初次发布
2	2018.01.16	V1.1	1、修改了个别语言描述 2、修改了模拟量 IO 点，即支持浮点数据
3	2018.03.05	V1.1.2	1、增加了心跳检测位(对外)
4	2019.12.26	V1.2 V1.4	1、IO 变量新建方法更新 2、IO 指令更新 3、增加了通信操作示例
5	2020.01.13	V1.5	1、修改不正确的描述
6	2022.01.10	V1.6	寄存器数量扩增为 1500, [1,100]仍为原有定义，[101,1500] 寄存器可在内部 PLC 中读写 1、增加 2.4 ModbusTCP 指令章节 2、增加 2.5 Multiprog 中读写寄存器章节 3、修改 ModbusTCP 点表
7	2023.07.11	V1.7	1、新增第 4 章节精简命令机制使用说明
8	2024.02.27	V2.0	1、重新拟定第 3、4、5 章节，重新定义 ModbusTCP 标准点表

目 录

前言.....**4**

第 1 章 功能简介.....**5**

第 2 章 ModBusTcp 协议介绍.....**5**

2.1 ModBusTCP 协议报文.....5

2.2 ModBusTCP 协议功能码.....6

2.3 ModBusTCP 接口配置.....6

2.4 ModBusTCP 指令.....7

2.4.1 GetModConState.....7

2.4.2 ReadModbusReg.....7

2.4.3 WriteModbusReg.....8

2.5 Multiprog(PLC)中读写寄存器.....8

2.5.1 ER_RobModbusCnctState.....9

2.5.2 ReadModbusReg.....9

2.5.3 WriteModbusReg.....10

第 3 章 接口调试.....**11**

3.1 ModScan 调试助手.....11

3.2 调试注意事项.....13

第 4 章 ModBusTCP 点表功能.....**15**

4.1 IO 功能.....15

4.1.1 物理 **DI** 与物理 **DO** 交互示例.....15

4.1.2 虚拟 **DI** 与虚拟 **DO** 交互示例.....15

4.1.3 虚拟 **AI** 与虚拟 **AO** 交互示例.....16

4.2 状态反馈功能.....18

4.2.1 机器人通讯心跳.....18

4.2.2 机器人状态反馈.....19

4.2.3 机器人位姿反馈.....20

4.3 命令功能.....21

4.3.1 上/下励磁.....21

4.3.2 启动程序运行.....22

4.3.3 暂停程序运行.....23

4.3.4 复位.....24

4.3.5 切换系统模式.....24

4.3.6 远程加载指定程序.....26

4.3.7 注销工程 26
4.3.8 设置全局速度 27
4.3.9 重置指令状态机 28
4.3.10 写入当前 **P** 变量 28
4.3.11 读取当前 **P** 变量 30
4.3.12 复位程序指针至首行 31
4.3.13 重新加载远程自启动程序 31
4.3.14 读取点位数组 **P** 32
4.3.15 示教当前 **P** 变量 33
4.3.16 设置当前点动坐标系 34
4.3.17 设置当前工具坐标系 35
4.3.18 设置当前用户坐标系 35
4.4 运动功能 36
4.4.1 点动/寸动功能 36
4.4.2 直接运动功能 37
第 5 章 ModBusTCP 点表 39

前言

概述

本手册适用于控制系统 RCS2 V1.33 及以上版本，描述埃斯顿二代控制器 ModBusTcp 协议接口功能介绍，包括功能简介、协议介绍、接口调试和功能点表。

读者对象

本手册仅供受过培训，熟悉各种适用国家标准的“控制、自动化和驱动工程”领域专业人员。

系统生产商：对自动化系统功能设计的技术人员。

系统集成商：指自动化设备集成的技术人员。

注意事项

在安装和调试这些组件时，操作人员必须严格遵循本文档的说明和解释。

相关负责人员必须确保所述产品的应用或使用满足所有安全要求，包括相关法律、法规、准则和标准。

尽管本文档经过精心编制，但由于其中所描述的产品仍处于不断更新换代中，我们可能不会在每次更新后都检查文档中所描述的产品性能数据、标准或其它特性总是与实际产品相一致。

本文档中难免会出现一些技术或者编辑错误，我们保留随时对文档信息做出修改之权力，恕不另行通知。对于已经变更的产品，如果本文档中的数据、图表以及文字描述没有修改，我们将不再特别加以声明。

任何人不得对软、硬件配置进行文本档中规定之外的修改，ESTUN 公司对因此而造成的一切后果不承担任何责任。

本文档中出现图示单位在没有特别标注说明时，默认单位为毫米 **mm**。

安全说明

警告	受伤的危险 不遵守本标志相关的安全说明将危及个人生命和健康安全。
注意	对环境和设备有危险 不遵守本标志相关安全说明可能明显危害环境和设备安全。
说明	说明或提示 该标志表示这些信息能够帮助您更好的理解安全说明。

第 1 章 功能简介

ModBusTcp 协议接口是指外部逻辑控制器（PLC 等）通过标准工业总线协议（ModBusTCP）与机器人通讯，读写机器人的虚拟 IO 端口的一种通讯方式。机器人可通过 ModBusTCP 的方式与外部设备进行交互，而无需额外扩展 IO 端口。

第 2 章 ModBusTcp 协议介绍

ModBus 是目前应用最广泛的现场总线之一，1999 年又推出了以太网运行的 ModBusTCP(工业以太网协议)。ModBusTCP 以一种比较简单的方式将 ModBus 帧嵌入 TCP 帧中。IANA 给 ModBus 协议赋予 TCP 端口号 502，这是其他工业以太网协议所没有的。ModBusTCP 还拥有其它工控设备的支持，如工业 人机界面、变频器、软启动器、电动机控制中心、以太网 I/O、各种现场总线的网桥等。

2.1 ModBusTCP 协议报文

ModBusTCP 采用 TCP/IP 和以太网协议来传输 ModBus 信息，因此与 ModBus 串行链路数据单元类似，ModBusTCP 的应用数据单元就是将 ModBus 简单协议数据单元（PDU）按照 TCP/IP 协议进行封装。一个 TCP 帧只能传送一个 ModBusADU，建议不要在同一个 PDU 中发送多个 ModBus 请求或相应。

以下表格为 ModBusTCP 的请求、应答报文：

请求：	传输标志	协议标志	长度	从站号	功能码	寄存器起始地址	寄存器长度
长度 (Byte)	2	2	2	1	1	2	2
应答：	传输标志	协议标志	长度	从站号	功能码	长度	数据值…

长度 (Byte)	2	2	2	1	1	1	

2.2 ModBusTCP 协议功能码

ModBus 的数据结构将数据区分为以下四种：

保持线圈(Holding Coils)	可读可写的 BOOL 型变量
输入线圈(Input Coils)	只读的 BOOL 型变量
保持寄存器(Input Registers)	可读可写的 WORD 型（16-bit）型变量
输入寄存器(Holding Registers)	只读的 WORD 型（16-bit）型变量

埃斯顿 ModBusTCP 通讯接口目前只支持读写保持寄存器，即以下功能码：

功能码	描述
0x03	读保持寄存器
0x06	写单个保持寄存器
0x10	写多个保持寄存器

2.3 ModBusTCP 接口配置

ER 系列机器人的 ModBusTCP 接口配置如下：

IP：192.168.6.68（实时系统 IP 地址）

Port：502

采集周期>50ms

响应超时>150ms

说明

服务器通过检测客户端的请求来判断连接是否存在。若与客户端连接后，超过 5 秒未接收到客户端的请求，服务器将自动断开连接并重新监听 502 端口。

ER 系列机器人在 ModBusTCP 保持寄存器中封装了一层数据结构

	地址	定义

Rob Send	40001~40250	机器人发送区域
Rob Receive	40251~40500	机器人接收区域
General	40501	用户用（寄存器已开放到内部 PLC）
	…	
	…	
	…	
	41500	

2.4 ModBusTCP 指令

ModbusTCP 指令列表：

• ModbusTCP 指令	• GetModConState
	• ReadModbusReg
	• WriteModbusReg

2.4.1 GetModConState

该指令用于获取机器人与外界使用 modbus 通讯的连接状态。对指令参数进行说明：

GetModConState IsConnected

IsConnected：连接状态（Is Connected）

数据类型：BOOL 变量

参数含义：返回当前的连接状态。

示例 1：

```
GetModConState(BOOL0)
```

2.4.2 ReadModbusReg

该指令用于读取指定 Modbus 寄存器的值。对指令参数进行说明：

ReadModbusReg RegisterID RegisterValue

RegisterID：目标寄存器 ID 号（Target Modbus Register ID）

数据类型：int，取值范围为 101~1500。

参数含义：想要读取的寄存器 ID 号。

RegisterValue：目标寄存器值（Target Modbus Register Value）

数据类型：INT 变量

参数含义：返回读取后的寄存器值。

示例 1：

```
ReadModbusReg(101, INT0)    // 读取 101 寄存器的值
```

2.4.3 WriteModbusReg

该指令用于设置指定 Modbus 寄存器的值。对指令参数进行说明：

WriteModbusReg RegisterID RegisterValue

RegisterID：目标寄存器 ID 号（Target Modbus Register ID）

数据类型：int，取值范围为 101~1500。

参数含义：想要修改的寄存器 ID 号。

RegisterValue：目标寄存器值（Target Modbus Register Value）

数据类型：INT 变量

参数含义：想要修改的寄存器值。

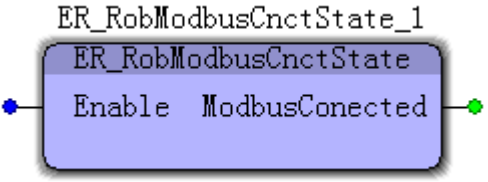
示例 1：

```
INT0.value = 100
WriteModbusReg(101, INT0)    // 将 101 寄存器的值改为 100
```

2.5 Multiprog(PLC)中读写寄存器

ER_ModbusTCP 功能库提供了读写 Modbus 寄存器的功能。可读写索引号为 101-1500 的寄存器的值，可用于与机器人控制器、外部 Modbus 设备进行数据交互。

2.5.1 ER_RobModbusCnctState



功能：获取标准 ModbusTCP 连接状态。

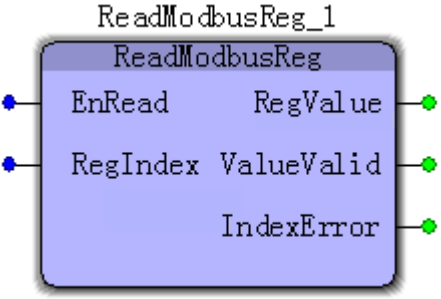
输入参数：

Enable(BOOL)：为 TRUE 期间持续获取 ModbusTcp 连接状态。

输出参数：

ModbusConcted(BOOL)：为 TRUE 表示 Modbus 已连接。

2.5.2 ReadModbusReg



功能： 读取指定 Modbus 寄存器的值。

输入参数：

EnRead（BOOL）：为 TRUE 期间持续获取寄存器的值。

RegIndex（INT）：寄存器 ID 号，支持 101-1500 范围内输入值。

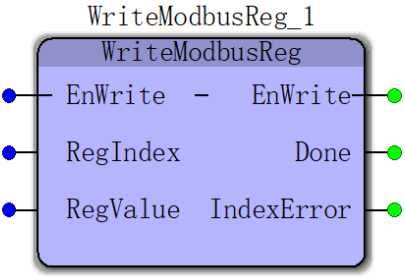
输出参数：

RegValue（WORD）：寄存器的值。

ValueValid（BOOL）：为 TRUE 表示输出值有效。

IndexError（BOOL）：为 TRUE 表示索引号超范围。

2.5.3 WriteModbusReg



功能： 写指定 Modbus 寄存器的值。

输入参数：

RegIndex（INT）：要写入的寄存器 ID，支持 101-1500 范围输入值。

RegValue（WORD）：待写入寄存器的值。

输出参数：

Done（BOOL）：为 TRUE 表示写入完成。

IndexError（BOOL）：为 TRUE 表示索引号超范围。

输入输出参数：

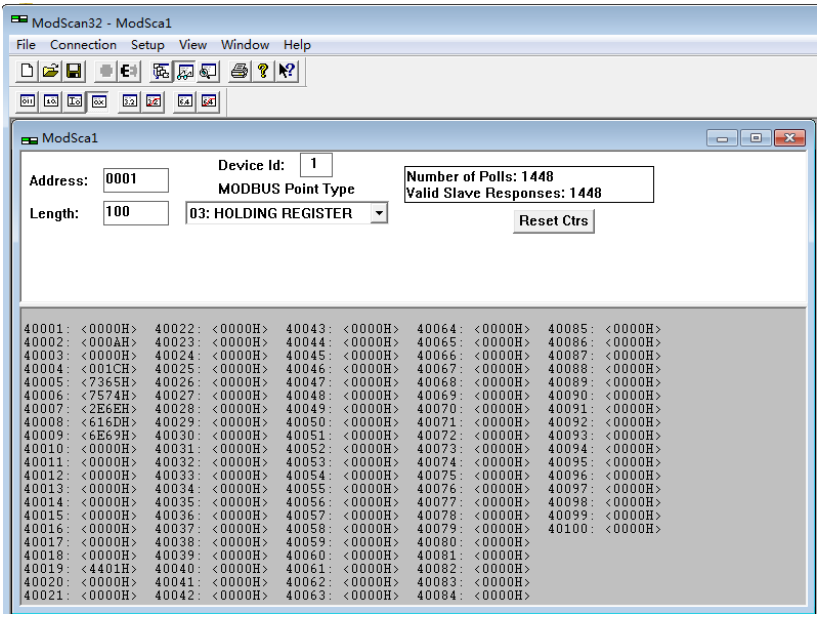
EnWrite（BOOL）：上升沿执行将值写入指定寄存器操作。

第 3 章 接口调试

该部分以 ModBus 调试助手 ModScan 为例，说明如何使用 ModBusTCP 通信接口与机器人交互，ModScan 模拟通信主站，机器人作为通信从站。

3.1 ModScan 调试助手

① 该调试软件界面如下:

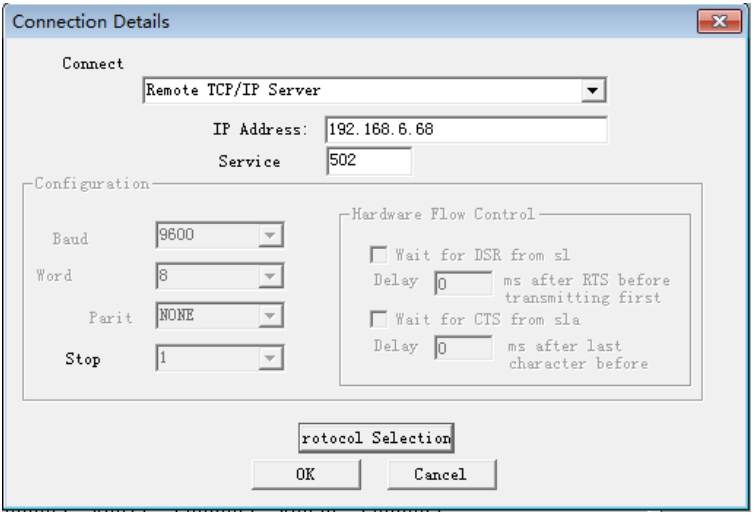


② 连接调试步骤

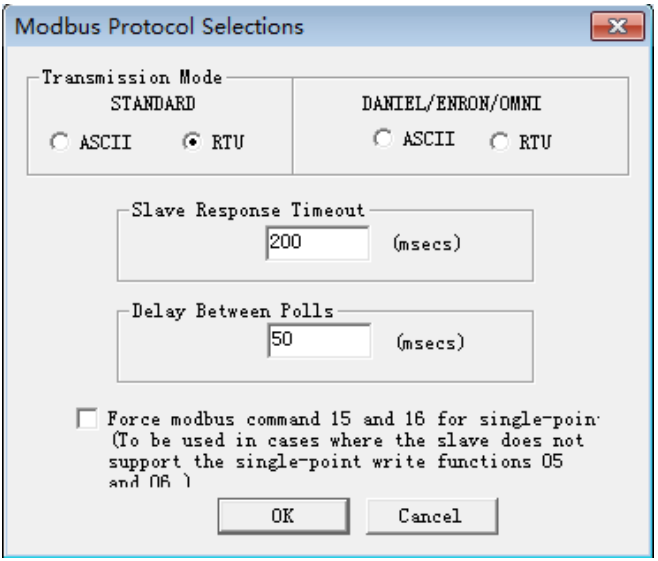
1、 将本地电脑连通机器人控制系统。(可 ping 通实时系统 IP192.168.6.68)

2、 配置 ModScan 连接参数

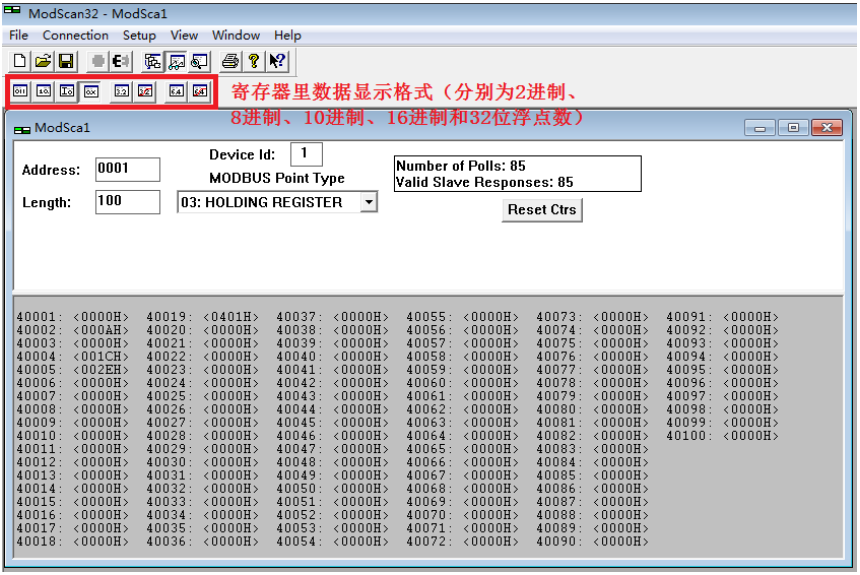
点击 Connection→Connect 进入如下界面，配置机器人 IP 和端口号。



先点击 protocol selection 配置连接参数，再点击 OK 完成配置并连接机器人



3、 Address（起始地址）设为 1，Length（长度）设为 100，功能码下拉选择 03：HOLDING REGISTER(保持寄存器)。



3.2 调试注意事项

- 默认机器人作为 ModbusTcp 通信从站，外部设备作为 ModbusTcp 通信主站，机器人只接受一个主站的连接，不可以多个客户端同时连接机器人（如有该需求请联系埃斯顿技术人员）；如果外部设备主站通信协议部分是由个人编制，可以先用 ModSim 测试功能可用后再去连接机器人。

- 通信交互时首先应确定两端设备通信寄存器地址是正确的，否则肯定不会成功。当外部设备给机器人发送数据时，可以先确定下 SimDI 对应的寄存器，因为从示教器端看 SimDI 信号接收情况最直观，当 SimDI 对应的外部设备通信地址确定下来时，可以根据机器人通信点表顺序偏移来确定其它的通信寄存器地址，当外部设备接收机器人数据时，同样可以通过 SimDO 来确定地址。

- 当与外部设备交互时，外部设备可以通过机器人返回的状态信息判断命令是否下发成功，在 40027 寄存里每个命令都有对应的成功标志位，此外，在通信过程中如果 40027 寄存器里错误位为 1，则需要用下发重置状态机命令去清除错误，否则后续的命令系统皆不响应。

- 当机器人与西门子 PLC 进行 ModBusTcp 通信交互模拟量数据时，由于西门子数据存储是根据低地址字节作为双字的高字节，高地址字节作为双字的低字节，所以在数据交互时需要特别注意对应关系。

```
//Read Robot Data to PLC
"Tag_1" := "NET_BUFF_1".MODBUS_READ[20]; //Robot 40021 -> PLC MW0
"Tag_2" := "NET_BUFF_1".MODBUS_READ[19]; //Robot 40020 -> PLC MW2
#x := DWORD_TO_REAL("Tag_3"); //MD0

//Write PLC Data to Robot
#y:=REAL_TO_DWORD("Tag_6");//MD4
"NET_BUFF_1".MODBUS_WRITE[16] := "Tag_4";//Write MW4 to 40069
"NET_BUFF_1".MODBUS_WRITE[15] := "Tag_5";//Write MW6 to 40068
```

第 4 章 ModBusTCP 点表功能

本章节主要介绍 ModbusTCP 标准点表的各项功能。主要内容包括：IO 功能介绍、状态反馈功能介绍、命令功能介绍、运动功能介绍。

注意此版本 ModbusTCP 点表与以前的点表功能不兼容，仅适用于控制系统 RCS2 V1.33 及以上版本，若实际使用时需要使用老版本的 ModbusTCP 功能，请联系厂家人员。

4.1 IO 功能

查看第 5 章中提供的 ModbusTCP 点表，当前支持读取物理 DI，读取物理 DO，读写虚拟 DI，读取虚拟 DO，读写虚拟 AI，读取虚拟 AO。

4.1.1 物理 DI 与物理 DO 交互示例

物理 DI，物理 DO 仅可读取，用于状态反馈，无法写入。

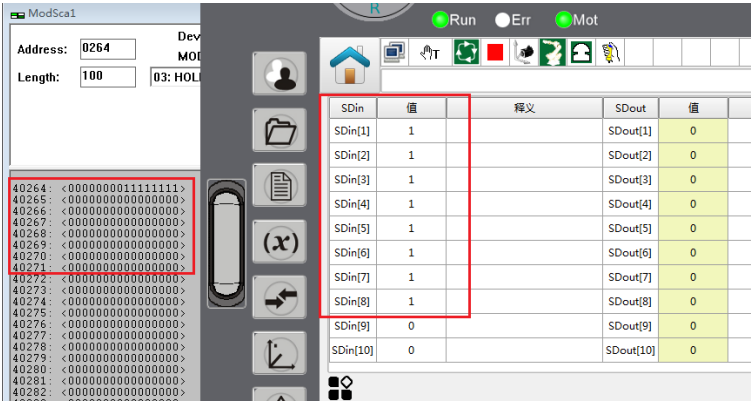
支持读取 1-32 路物理 DI，读取 1-32 路物理 DO，使用 ModScan 软件进行验证：

如下图所示，Din[4]使用仿真置为 1 后，40025 寄存器 bit3 的值同步显示为 1，Dout[3]和 Dout[4]输出为 1，40023 寄存器的 bit2 和 bit3 同步显示为 1。

4.1.2 虚拟 DI 与虚拟 DO 交互示例

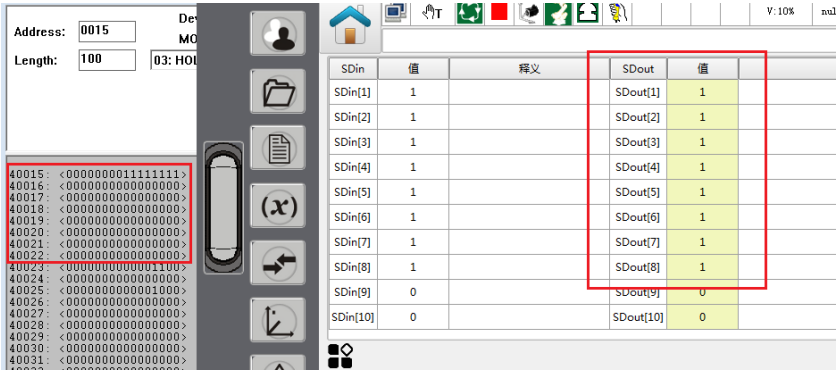
支持读写 1~128 路虚拟 DI，使用 ModScan 软件进行验证：

如下图所示，将 40264 寄存器的 bit0~7 置为 1，示教器上 SDin[1]~SDin[8]的值都显示为 1。



支持读取 1~128 路虚拟 DO，使用 ModScan 软件进行验证：

如下图所示，将示教器上 SDout[1]~SDout[8]的值置为 1。40015 寄存器的 bit0~7 都显示为 1。

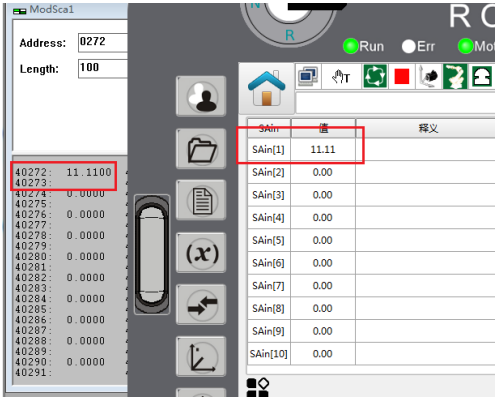


4.1.3 虚拟 AI 与虚拟 AO 交互示例

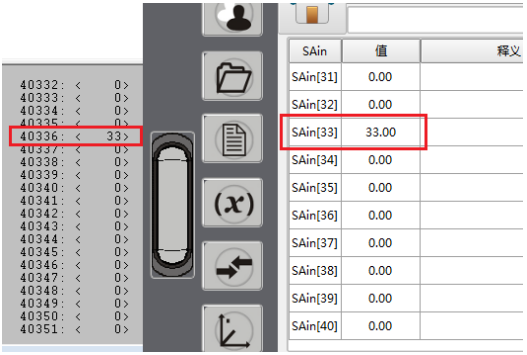
支持读写 1~64 路虚拟 AI，其中 1~32 路虚拟 AI 每一路使用两个寄存器，表示浮点型数据；33~64 路虚拟 AI 每一路使用一个寄存器，表示 INT 型数据。

使用 ModScan 软件进行验证：

如下图所示，使用 40272 和 40273 两个寄存器设置一个浮点数 11.11，示教器上 SAin[1]的值同步显示为 11.11。



如下图所示，将 40336 寄存器的值设置为 33，示教器上 SAin[33]的值同步显示为 33。

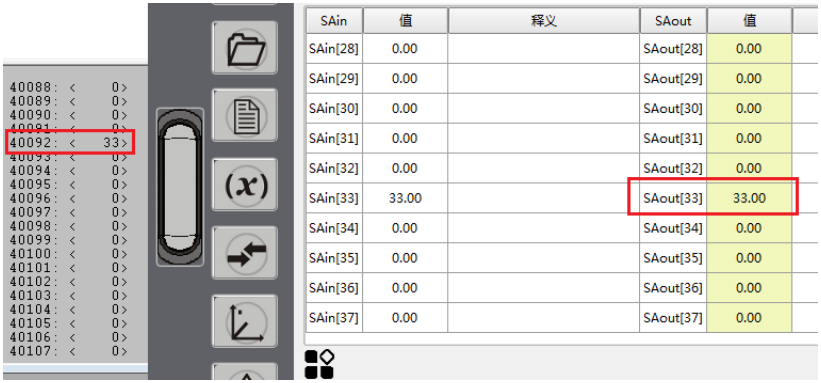


支持读取 1~64 路虚拟 AO，其中 1~32 路虚拟 AO 每一路使用两个寄存器，表示浮点型数据；33~64 路虚拟 AO 每一路使用一个寄存器，表示 INT 型数据。

使用 ModScan 软件进行验证：

如下图所示，在示教器上将 SAout[1]的值设置为 11.11，那么 40028 和 40029 两个寄存器显示一个浮点数 11.11。

如下图所示，在示教器上将 SAout[33]的值设置为 33，那么 40092 寄存器的值同步显示为 33。



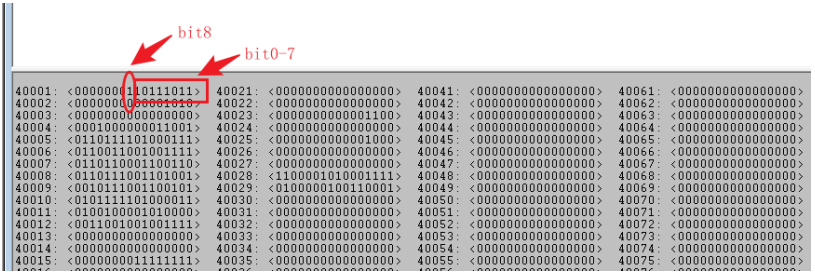
4.2 状态反馈功能

查看第 5 章中提供的 ModbusTCP 点表，表中给出了机器人系统的一些状态反馈，如通讯心跳，系统模式，励磁状态，系统运行状态，程序运行状态，机器人当前位姿等等。以下小节仅介绍部分状态，具体可详见第 5 章提供的点表。

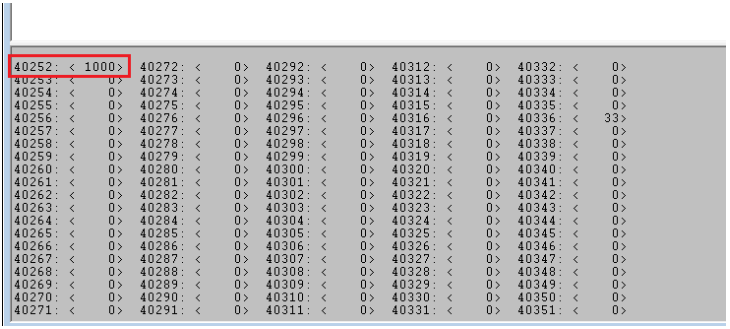
4.2.1 机器人通讯心跳

当前提供两种通讯心跳：累加计数与周期性变化。

使用 ModScan 软件进行验证，如下图所示，40001 寄存器的 bit0~7 会循环累加计数，而 bit8 则会 0/1 进行切换，每个信号都持续 500 毫秒。



bit8 的信号维持时间可通过 40252 寄存器进行设置，单位为毫秒，当设置的值小于 500 时，默认按照 500 处理。如下图所示，写入 1000 时，0/1 信号维持时间就变为 1000 毫秒。

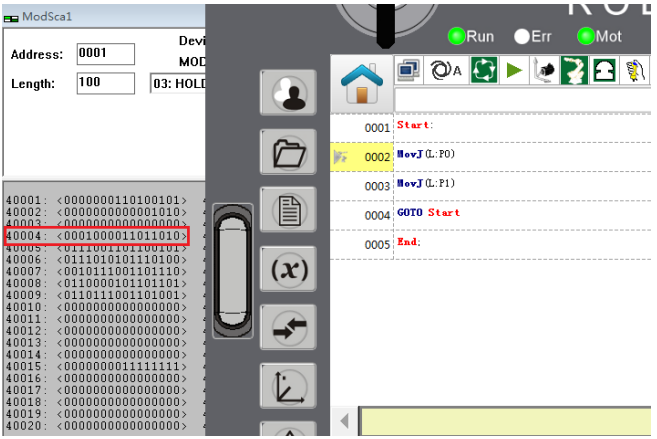


4.2.2 机器人状态反馈

可通过指定寄存器获取当前机器人系统的一些状态，查看点表可知：

如 40004 寄存器的 bit0~2 表示当前系统模式，bit3 表示使能状态，bit4 表示系统运行状态，bit5 表示错误状态，bit6 表示程序运行状态，bit7 表示机器人运动状态等等。

使用 ModScan 软件进行验证，如下图所示：



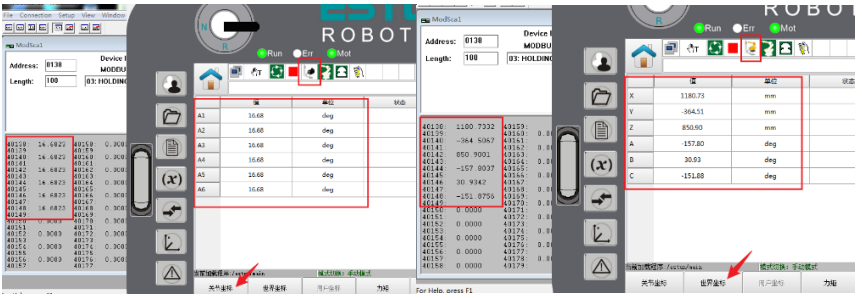
从寄存器的值可以看出，当前系统处于自动模式，上励磁，系统正常运行，程序处于运行状态，机器人处于运动状态，与示教器上的反馈状态是一致的。

4.2.3 机器人位姿反馈

查看第 5 章中提供的扩展 ModbusTCP 点表，可通过寄存器获取到机器人的位姿，具体为 40138~40156 寄存器。

寄存器反馈的值会依据当前点动坐标系类型发生改变，当点动坐标系为关节坐标系时，会显示关节坐标；当点动坐标系为笛卡尔坐标系时，会显示世界坐标。

使用 ModScan 软件进行验证，如下图所示：



4.3 命令功能

此版本 ModbusTCP 点表新增了许多交互命令，且命令实现机制作了改动。只要往指定寄存器中写入相应的值，即可触发命令。

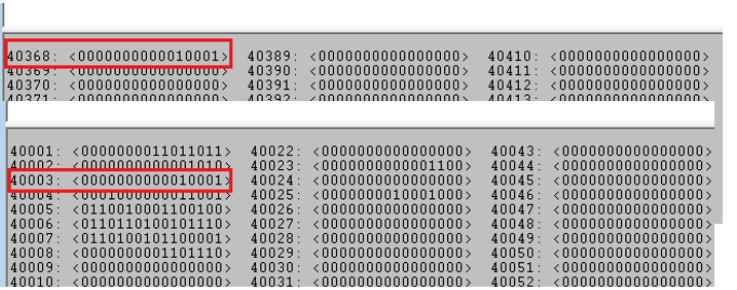
使用该机制给机器人发送命令操作，前提有三：

① 读写标志寄存器 40368 中需写入值 0x11，以打开命令下发权限；

② 命令执行状态寄存器 40027 寄存器的 bit13 值为 0；(当此 bit 位值为 1 时，表示有命令执行出错，需要通过 40251 寄存器进行复位)

③ 个别命令需在远程模式下才能生效。

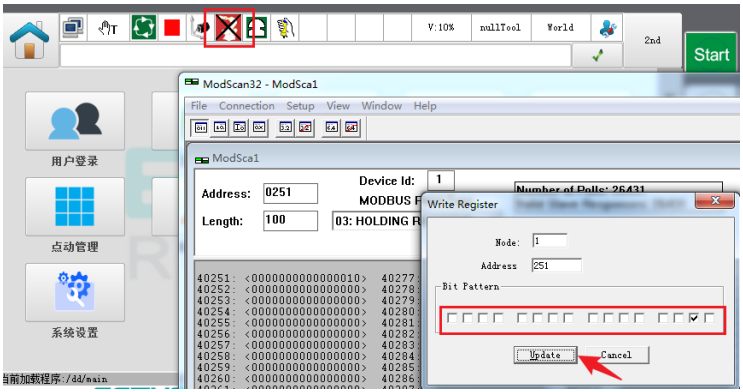
使用 ModScan 软件进行验证，参考 3.1 小节，在与控制器建立连接后。首先往 40368 寄存器里写入 0x11，40003 寄存器状态同步变为 0x11，此时命令交互权限被打开。接下来就可以通过赋值指定寄存器，执行相应的命令，并且执行成功后，可以观察 40027 的相应 bit 位或是各个状态寄存器的变化。



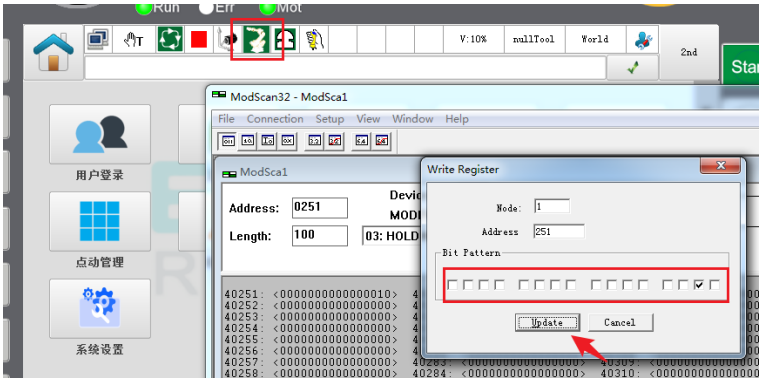
4.3.1 上/下励磁

使用 ModScan(PLC)给机器人发送上下励磁命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0(上下励磁命令全模式生效，但在远程模式下，需要将远程自动励磁置为 OFF)。

① 若当前未上励磁，往 40251 寄存器中写入值 2(bit1 置为 1)，即可触发上励磁命令，机器人将会上励磁。



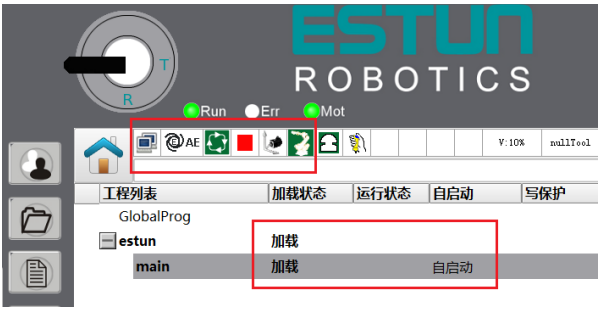
② 若当前已上励磁，往 40251 寄存器中写入值 2(bit1 置为 1)，即可触发下励磁命令，机器人将会下励磁。



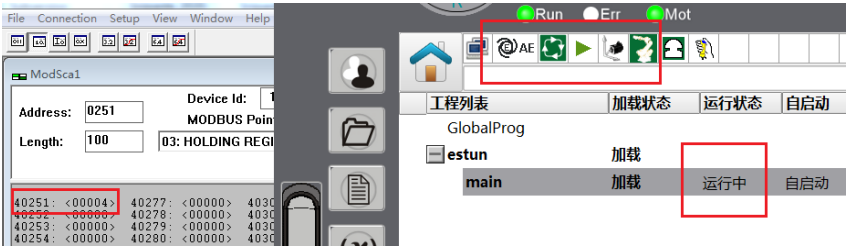
4.3.2 启动程序运行

使用 ModScan(PLC)给机器人发送启动程序运行命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0，当前为远程模式且程序为远程自启动。

① 将机器人示教器钥匙开关拨至远程状态，当前处于上励磁并加载已经设为自启动的程序，如下图所示。



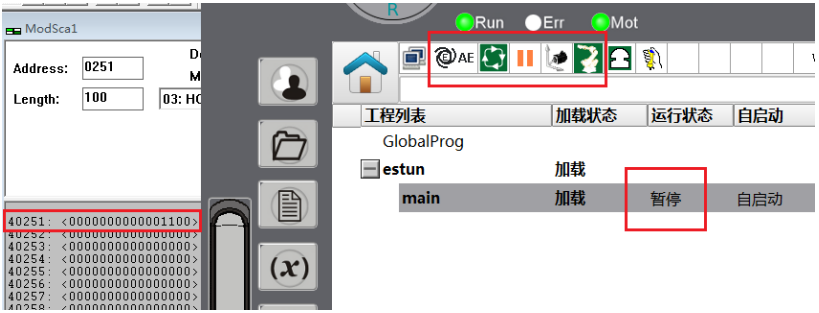
② 往 40251 寄存器中写入值 4(bit2 置为 1)，即可触发启动程序运行命令，如下图所示，程序开始运行。



4.3.3 暂停程序运行

使用 ModScan(PLC)给机器人发送暂停程序运行命令，暂停功能与其他命令有所区别，属于电平信号检测，不受命令权限影响，只要将 40251 寄存器的 bit3 置为 1，那么就会触发程序暂停，且此时通过其他方式启动程序，如果暂停信号没有复位，那么程序会再次被暂停(暂停信号全模式检测)。

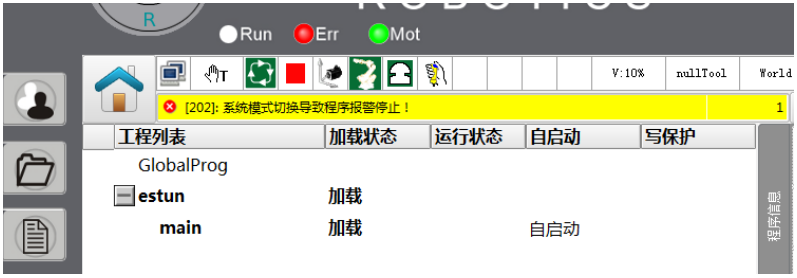
如下图所示，当程序处于运行状态时，触发暂停信号，程序被暂停。



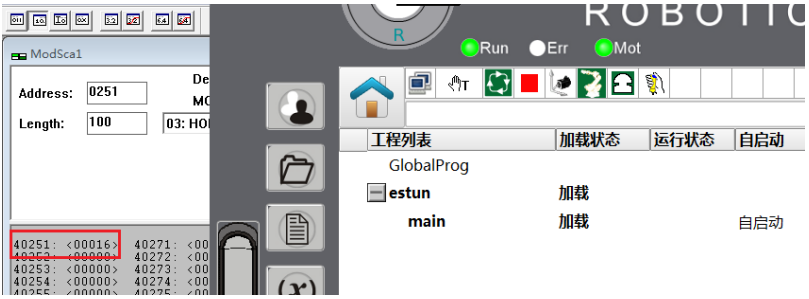
4.3.4 复位

使用 ModScan(PLC)给机器人发送清除所有报警命令(复位机器人)，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0(复位命令全模式生效)。

① 如下图所示，机器人系统有一条报警显示。



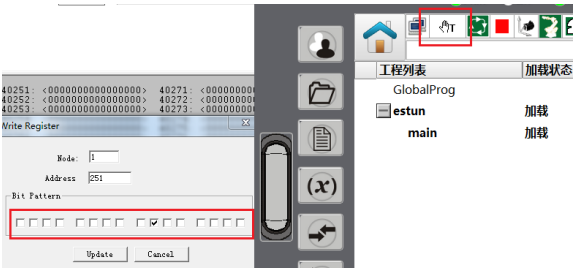
② 往 40251 寄存器中写入值 16(bit4 置为 1)，即可复位机器人，清除报警。



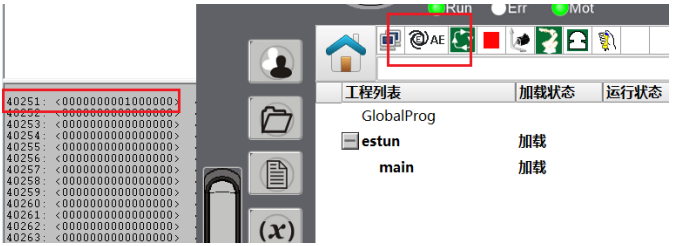
4.3.5 切换系统模式

使用 ModScan(PLC)给机器人发送切换系统模式的命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0。

① 若当前机器人处于手动模式，往 40251 寄存器中写入值 64(bit6 置为 1)，则系统会先切换到远程模式，若再次写入 64，则会切换到自动模式，后续写入 64 则会在远程与自动模式间切换，如下图所示。

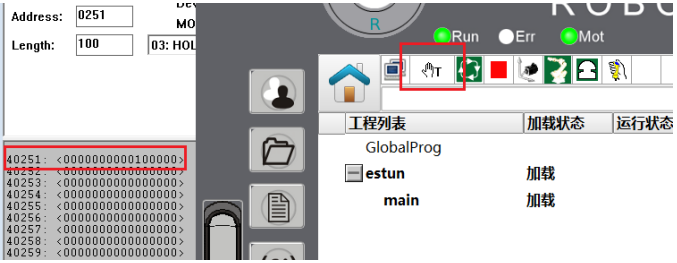


将 40251 寄存器的 bit6 置为 1，会先切换到远程模式。



再次置为 1，则会切换到自动模式。

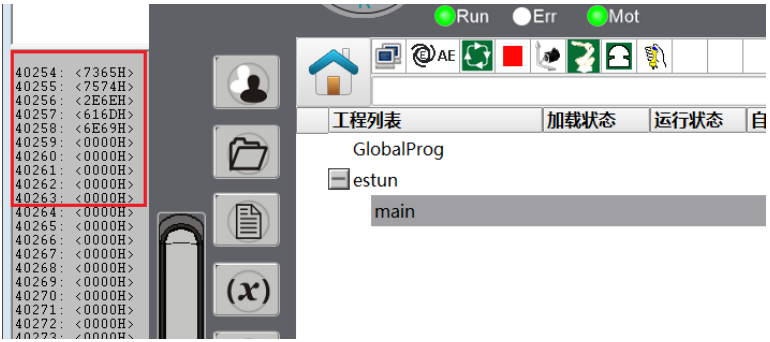
② 往 40251 寄存器中写入值 32(bit5 置为 1)，则会切换到手动模式。



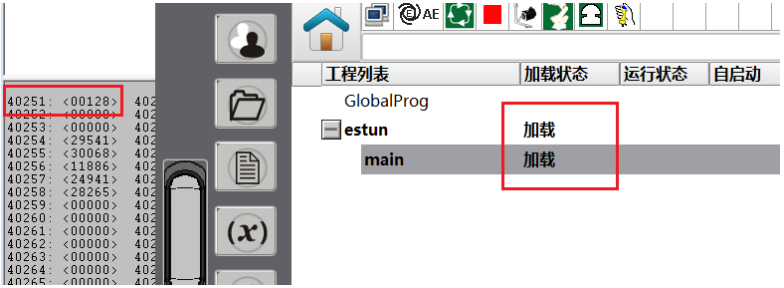
4.3.6 远程加载指定程序

使用 ModScan(PLC)给机器人发送加载程序命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0，加载程序名为 estun.main。

① 机器人程序名 estun.main 对应的 16 进制数值分别为 0x7365、0x7574、0x2E6E、0x616D 和 0x6E69，将上述数值分别存入以 40254 开头的 5 个寄存器内。



② 往 40251 寄存器中写入值 128(bit7 置为 1)，此时示教器上加载 estun.main。



4.3.7 注销工程

使用 ModScan(PLC)给机器人发送注销工程命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0。

① 往 40251 寄存器中写入值 256(bit8 置为 1)，此时示教器上加载的工程会被注销。

4.3.8 设置全局速度

使用 ModScan(PLC)给机器人发送设置全局速度的命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0。

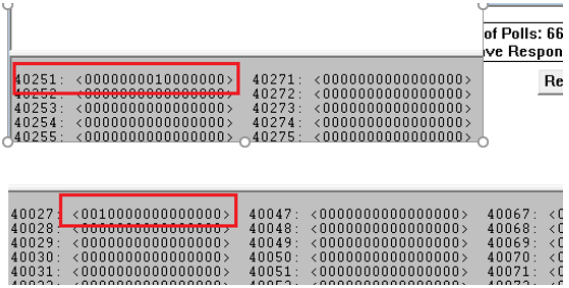
① 首先往 40253 寄存器中写入待设置的全局速度值，如下图所示以 99 为例。

② 往 40251 寄存器中写入值 512(bit9 置为 1)，此时示教器上全局速度被设置为 99%。

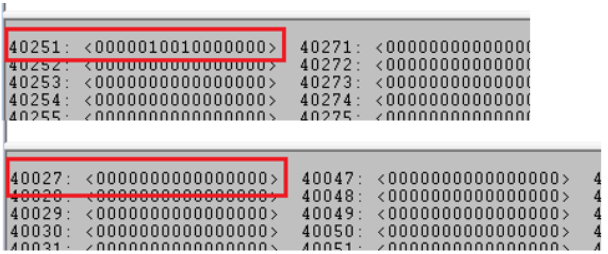
4.3.9 重置指令状态机

使用 ModScan(PLC)重置指令执行状态机，当各命令执行错误，如执行加载程序时，程序名对应的寄存器为空，此时 Modbus 状态寄存器 40027 的 bit13 会被置 1，当此位置 1 时，系统是不响应任何命令的，需要通过 40251 寄存器的 bit10 置 1 来复位。

① 加载程序命令执行出错，40027 的 bit13 会被置 1。



② 上图执行出错后，需要赋值 40251 的 bit10 来进行复位，此时控制器才会继续响应其他 Modbus 命令。

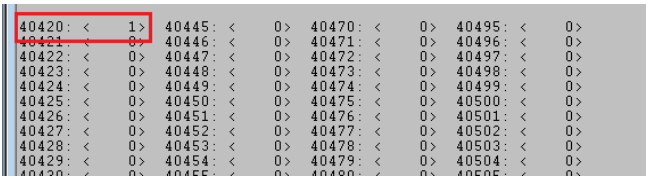


4.3.10 写入当前 P 变量

使用 ModScan(PLC)给机器人发送写入当前 P 变量命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0。

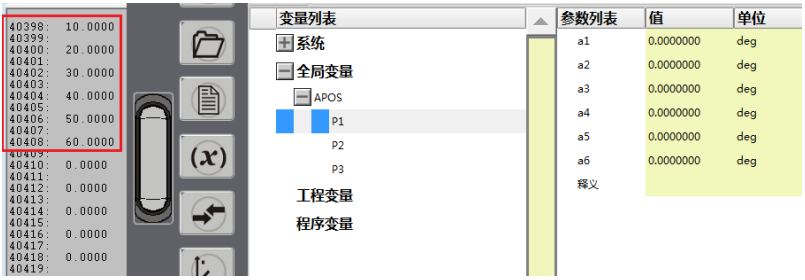
具体使用方法为，首先在 40420 寄存器中写入 P 变量的序号，如写入 1.2.3…就表示变量 P1.P2.P3…(目前仅支持全局变量，且需要在示教器上预先建好)，然后在 40398~40417 寄存器中写入变量的坐标值、Cfg 值以及点位类型，最后往 40251 寄存器中写入命令对应的值触发命令。

① 往 40420 寄存器中写入 1，表示将要操作的变量为 P1。

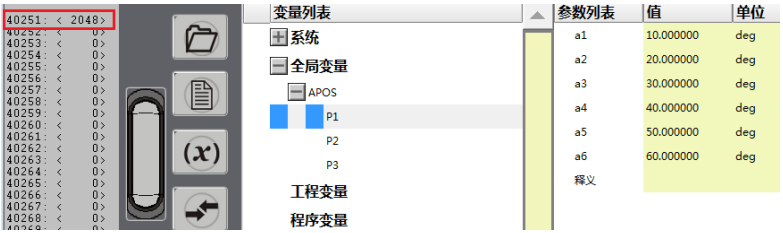


② 往 40398~40417 寄存器中写入变量的坐标值、Cfg 值以及点位类型，查看点表可知，点位的坐标值都是用两个寄存器表示的浮点型数据，Cfg 值与点位类型是单个寄存器表示，注意点位类型寄存器 40417 的值不能写错(0 表示 APOS，1 表示 CPOS)。如果在示教器上 P1 是 APOS，但是 40417 的值为 1，那么触发写入命令后，会将 P1 强制写成 CPOS。

如下图所示，将 APOS 点 P1 的值设置为图中所示的值(40410~40417 的值均为 0)。



③ 往 40251 寄存器中写入值 2048(bit11 置为 1)，此时将设置的值写入 P1 变量。

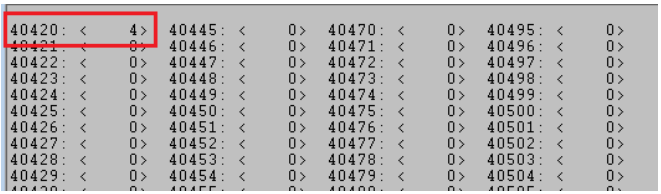


4.3.11 读取当前 P 变量

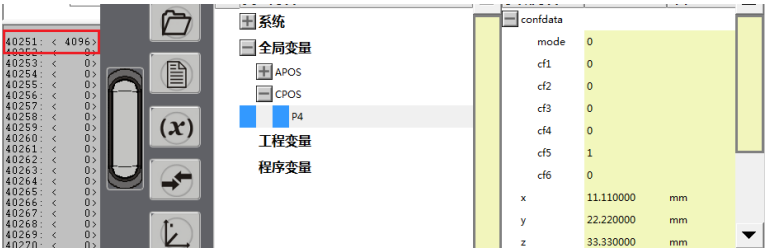
使用 ModScan(PLC)给机器人发送读取当前 P 变量命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0。

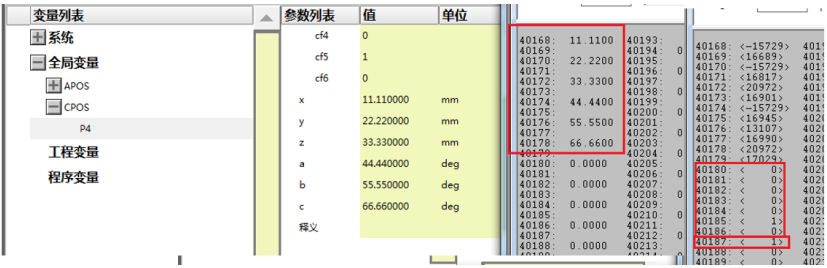
具体使用方法为，首先在 40420 寄存器中写入 P 变量的序号，如写入 1.2.3…就表示变量 P1.P2.P3…(目前仅支持全局变量，且需要在示教器上预先建好)，然后往 40251 寄存器中写入命令对应的值触发命令，那么最终读取到的坐标值、Cfg 值以及点位类型会在寄存器 40168~40187 中显示。

① 往 40420 寄存器中写入 4，表示将要操作的变量为 P4。



② 往 40251 寄存器中写入值 4096(bit12 置为 1)，此时将会读取 P4 变量的坐标值、Cfg 值以及点位类型。

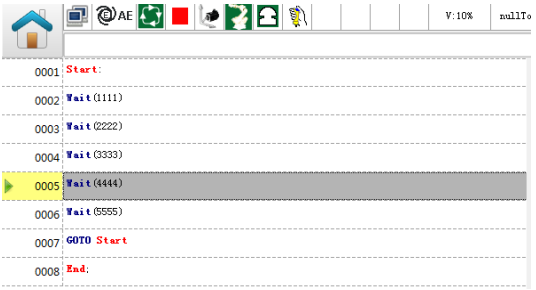




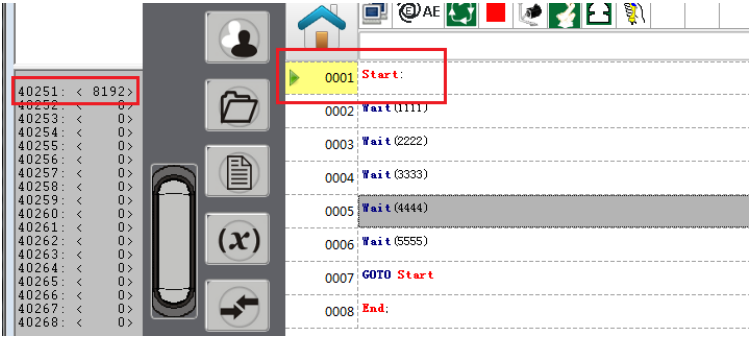
4.3.12 复位程序指针至首行

使用 ModScan(PLC)给机器人发送复位程序指针命令，使 PC 指针回到第一行(此命令不会注销程序)。确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0，且当前为远程模式。

① 如下图所示，将当前程序 PC 指针停留在第 5 行。



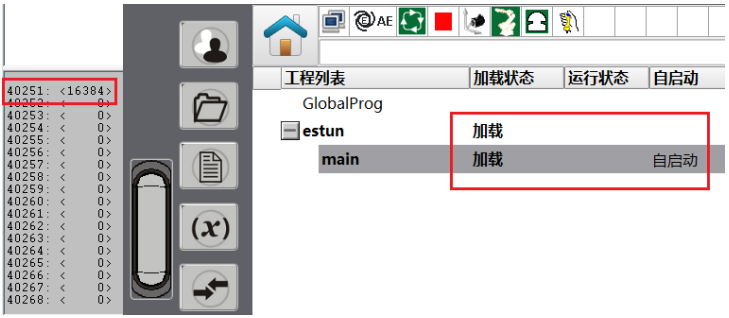
② 往 40251 寄存器中写入值 8192(bit13 置为 1)，触发设置 PC 命令，PC 指针回到第一行。



4.3.13 重新加载远程自启动程序

使用 ModScan(PLC)给机器人发送重新加载远程自启动程序命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0，且当前为远程模式。

往 40251 寄存器中写入值 16384(bit14 置为 1)，即可触发重新加载远程自启动程序命令(同时也会复位 PC 指针)。

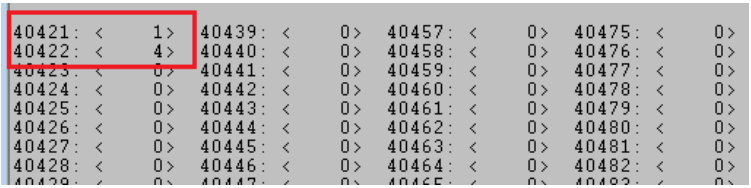


4.3.14 读取点位数组 P

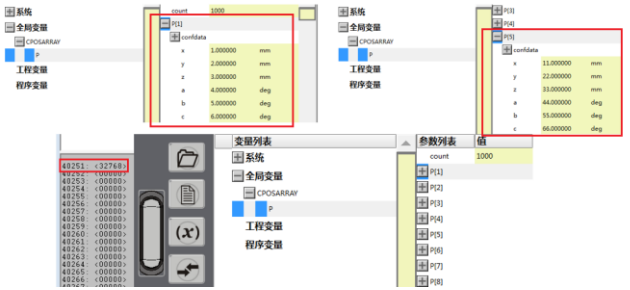
使用 ModScan(PLC)给机器人发送读取点位数组 P 的命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0。

具体使用方法为，首先需要在 40421 和 40422 寄存器中，分别写入点位数组 P 的序号和偏移，如在 40421 中写入 1，40422 中写入 4。那么就表示将要读取点位数组变量 P[1]和 P[5](该序号值为 1+4，目前仅支持全局变量，需要在示教器上预先建好 P 变量)，然后往 40251 寄存器中写入命令对应的值触发命令，那么最终读取到 P[1]的坐标值、Cfg 值以及点位类型会在寄存器 40190~40209 中显示；P[5]的坐标值、Cfg 值以及点位类型会在寄存器 40212~40231 中显示。

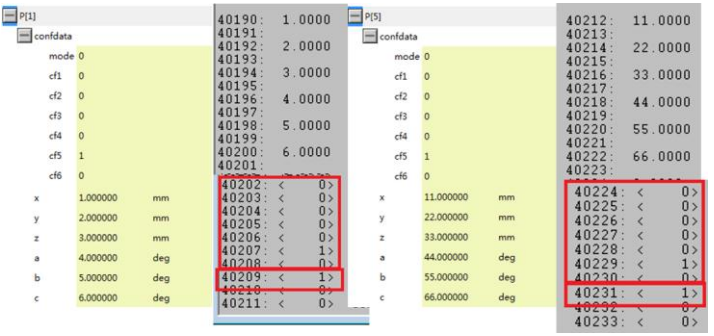
① 往 40421 寄存器中写入 1，往 40422 寄存器中写入 4，表示将要操作的变量为 P[1]和 P[5]。



② 往 40251 寄存器中写入值 32768(bit15 置为 1)，此时将会读取 P[1]和 P[5]变量的坐标值、Cfg 值以及点位类型。下图中以坐标系位置数组为例(CPOSARRAY)，关节位置数组(APOSARRAY)同理。



③ 最终读取到的值如下图所示。(P[1]和 P[5]的点位类型都是 CPOS)

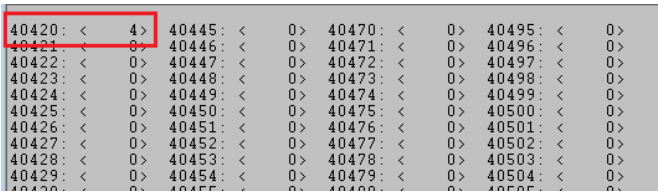


4.3.15 示教当前 P 变量

使用 ModScan(PLC)给机器人发送示教当前 P 变量命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0。

具体使用方法为，首先在 40420 寄存器中写入 P 变量的序号，如写入 1.2.3…就表示变量 P1.P2.P3…(目前仅支持全局变量，且需要在示教器上预先建好)，然后将 40382 寄存器的 bit1 置 1 触发命令，那么最终会将机器人当前坐标值写入当前 P 变量(会自动识别点位类型)。

① 往 40420 寄存器中写入 4，表示将要操作的变量为 P4。



② 往 40382 寄存器中写入值 1(bit1 置为 1)，触发命令，由于该寄存器为复归型寄存器，即赋值为 1 后，命令执行完成，又会复位为 0。

4.3.16 设置当前点动坐标系

使用 ModScan(PLC)给机器人发送设置点动坐标系命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0。

具体使用方法为，往 40383 寄存器中写入不同的值来触发设置不同的点动坐标系，0 表示关节、1 表示世界、2 表示工具、3 表示用户。如下图所示，点动坐标系开始为关节坐标系，当写入 1 时，会切换到世界坐标系。

4.3.17 设置当前工具坐标系

使用 ModScan(PLC)给机器人发送设置工具坐标系命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0。

具体使用方法为，往 40384 寄存器中写入不同的值来触发设置相应的工具坐标系，工具坐标系变量需提前在示教器上建好，且名称必须依照 TOOLx 的形式(x 表示数字)。如当往 40384 寄存器中写入 3 时，会设置 TOOL3，即写入值的就表示 TOOL 变量名后缀的数字。而当写入 0 时，则会重置为 nullTool。

4.3.18 设置当前用户坐标系

使用 ModScan(PLC)给机器人发送设置用户坐标系命令，确保命令权限已打开，状态寄存器 40027 的 bit13 位为 0。

具体使用方法为，往 40385 寄存器中写入不同的值来触发设置相应的用户坐标系，用户坐标系变量需提前在示教器上建好，且名称必须依照 USERCOORx 的形式(x 表示数字)。如当往 40385 寄存器中写入 3 时，会设置 USERCOOR3，即写入值的就表示 USERCOOR 变量名后缀的数字。而当写入 0 时，则会重置为 World。

4.4 运动功能

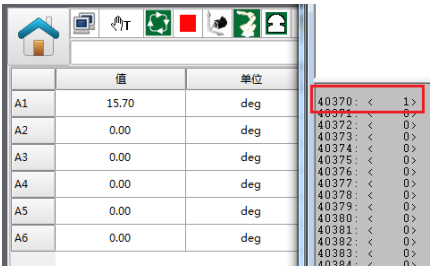
查看第 5 章中提供的 ModbusTCP 点表，当前点表支持连续点动或寸动机器人，以及控制机器人直接运动到指定位置点。

4.4.1 点动/寸动功能

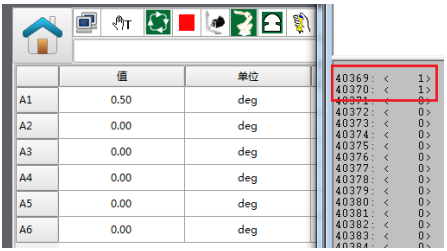
使用 ModScan(PLC)执行点动/寸动机器人功能，确保此时处于上励磁状态，系统模式为手动模式。

具体使用方法为，通过操作 40370~40381 共 12 个寄存器，实现机器人的点动与寸动，其中 40370~40375 表示往机器人正限位方向运动，40376~40381 表示往机器人负限位方向运动。

① 往 40370 寄存器中写入 1(bit1 置 1)，则会开始点动机器人的 J1 轴(注：此时点动坐标系是关节坐标系，如果设置为世界坐标系，则会往 X 正方向运动)，复位为 0 时机器人则会停止。



② 当需要进行寸动时，将 40369 寄存器的值设为 1，同时可通过 40394~40395 以及 40396~40397 寄存器，设置寸动关节步长于寸动线性步长。(注：两个值均为浮点型数据，用两个寄存器表示。寸动关节步长取值范围为 0.0~1.0°，默认取值 0.5；寸动线性步长范围为 0.0~10.0mm，默认取值 1.0)



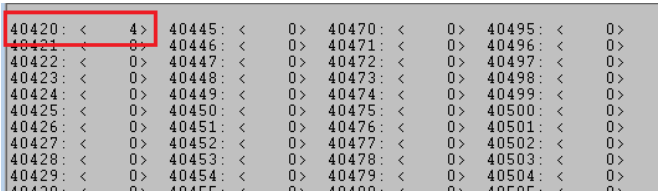
注：(1)所有点动寄存器均为沿信号检测，当值 0->1 时机器人运动，当值 1->0 时机器人停止；(2)点动寄存器每次只能触发一个，当触发某个点动寄存器时，需确保其他寄存器的值为 0。

4.4.2 直接运动功能

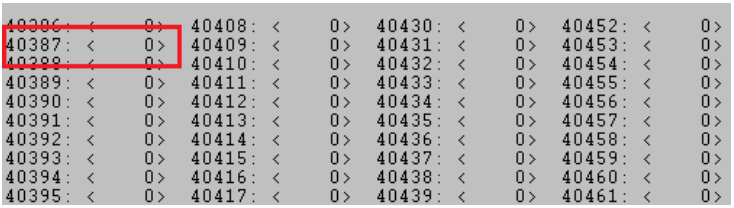
使用 ModScan(PLC)执行直接控制机器人运动功能，确保此时处于上励磁状态，系统模式为手动模式。

具体使用方法为：

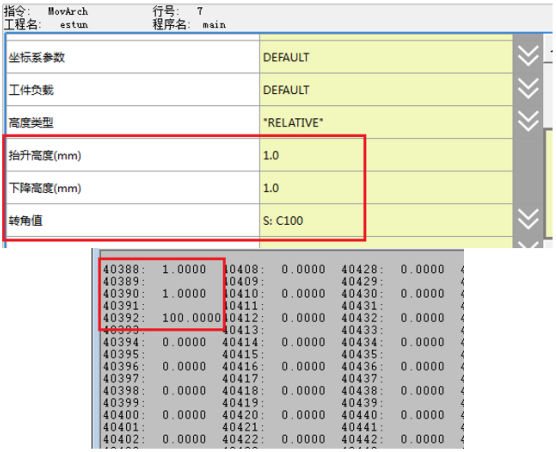
① 首先需要设置要运动到的点位，此点位就是 6.3.10 及 6.3.11 小节中涉及的 P 变量，如往 40420 寄存器中写入 4，就表示最终会往 P4 变量运动。



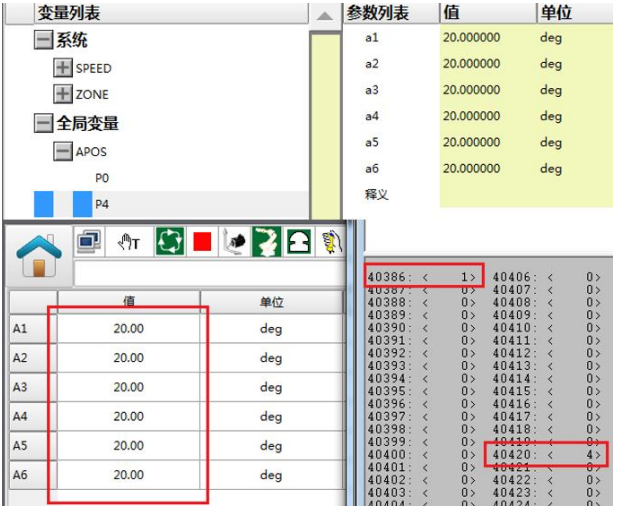
② 然后需要设置 40387 寄存器的值，该寄存器表示直接运动方式：0 表示 MovJ、1 表示 MovL、2 表示 MovArch、3 表示 MovLArch。



③ 如果当前运动方式为 MovArch 或者 MovLArch(注：仅 Scara 机型支持此种运动方式)，还需设置 40388~40389 寄存器(抬升高度)、40390~40391 寄存器(下降高度)、40388~40389 寄存器(转角值)。类比示教器上 MovArch 或 MovLArch 指令的这三个值。



④ 最后将 40386 寄存器的 bit1 值为 1，触发直接运动功能，那么机器人会持续运动到指定点位。



注：(1)直接运动功能寄存器为沿信号检测，当值 0->1 时触发运动，当值 1->0 时会停止运动；(2)当运动到位后，需将运动功能寄存器 40386 复位为 0，以复位机器人。

第 5 章 ModBusTCP 点表

Robot 作为 ModbusTCP 通信的 Server，提供了 1500 个寄存器地址用于读写；

40001~40250 为发送区域(用于输出 Robot 相关状态信息给到外部 PLC，只读)；

40251~40500 为接收区域(用于外部 PLC 对 Robot 写入相关操作指令和信息)；

40501~41500 为寄存器通用读写区域，用户可自定义(支持在 Multiprog 和指令中读写)。

Robot To 外部 PLC			
寄存器	**定义**	**说明**	**注释**
40001	心跳检测寄存器(提供两种跳检测方式)	bit0~7 位：循环累加	该值由 1 到 256 循环
		bit8 位：周期性变化	以 500ms 为周期进行 0/1 切换(周期可通过寄存器进行调整)
40002	当前全局速比	当前全局速比	值为百分比
40003	命令权限反馈	显示 40368 寄存器的值	0x11 表示打开命令权限
40004	Robot 状态信息	bit0：手动操作模式	当状态满足时，对应 bit 位会置为 1
		bit1：自动操作模式	
		bit2：远程操作模式	
		bit3：使能状态	
		bit4：系统运行状态	
		bit5：系统错误状态	
		bit6：程序运行状态	
		bit7：机器人运动状态	
		bit8：预留	
		bit9：预留	
		bit10：系统报警状态	
		bit11：伺服报警状态	
		bit12：急停状态	
		bit13：安全门状态	
		bit14：预留	
		bit15：预留	
40005 ~ 40014	当前加载的工程名.程序名	以 ASCII 码的形式显示"工程名.程序名",共 10 个寄存器最大显示 20 个字节	如 estun.main 对应的 16 进制数值分别为 0x7365、0x7574、0x2E6E、0x616D 和 0x6E69

40015	SimDO[1-16]	1~16 路虚拟 DO	显示虚拟 DO，寄存器的每个 bit 代表一路虚拟 DO(只读)
40016	SimDO[17-32]	17~32 路虚拟 DO	
40017	SimDO[33-48]	33~48 路虚拟 DO	
40018	SimDO[49-64]	49~64 路虚拟 DO	
40019	SimDO[65-80]	65~80 路虚拟 DO	
40020	SimDO[81-96]	81~96 路虚拟 DO	
40021	SimDO[97-112]	97~112 路虚拟 DO	
40022	SimDO[113-128]	113~128 路虚拟 DO	
40023	DO[1-16]	1~16 路物理 DO	物理 DO 反馈(只读)
40024	DO[17-32]	17~32 路物理 DO	
40025	DI[1-16]	1~16 路物理 DI	物理 DI 反馈(只读)
40026	DI[17-32]	17~32 路物理 DI	
40027	Rob 执行命令状态	bit0：预留	命令执行是否成功的状态反馈
		bit1：预留	
		bit2：启动程序执行成功	
		bit3：停止程序执行成功	
		bit4：复位命令执行成功	
		bit5：上使能执行成功	
		bit6：下使能执行成功	
		bit7：加载程序执行成功	
		bit8：注销工程执行成功	
		bit9：设置全局速度执行成功	
		bit10：等待控制权(不使用)	
		bit11：等待命令(不使用)	
		bit12：等待命令执行完成(不使用)	
		bit13：命令执行错误	
		bit14：复位程序指针执行成功	
		bit15：预留	
40028 ~ 40059	SimAO[1-16]	1~16 路虚拟 AO	每两个寄存器对应一个模拟量，分别表示高低位，如 40028 表示 SimAO[1] 的低位，40029 表示 SimAO[1] 的高位(只读)
40060 ~ 40091	SimAO[17-32]	17~32 路虚拟 AO	
40092 ~ 40107	SimAO[33-48]	33~48 路虚拟 AO	每一个寄存器对应一个模拟量，用于传输 INT 类型数据，如 40092 表示 SimAO[33](只读)
40108 ~ 40123	SimAO[49-64]	49~64 路虚拟 AO	

40124	轴报警状态	bit0~7：预留	
		bit8：J1 轴伺服报警	有报警时,对应 bit 位会置为 1
		bit9：J2 轴伺服报警	
		bit10：J3 轴伺服报警	
		bit11：J4 轴伺服报警	
		bit12：J5 轴伺服报警	
		bit13：J6 轴伺服报警	
		bit14：J7 轴伺服报警	
		bit15：J8 轴伺服报警	
40125	轴报警代码	J1 轴伺服报警 ID	显示每个轴的报警 ID 号
40126		J2 轴伺服报警 ID	
40127		J3 轴伺服报警 ID	
40128		J4 轴伺服报警 ID	
40129		J5 轴伺服报警 ID	
40130		J6 轴伺服报警 ID	
40131		J7 轴伺服报警 ID	
40132		J8 轴伺服报警 ID	
40133	当前点动坐标系	0:关节，1:世界，2:工具，3:用户	
40134	当前工具坐标系号	显示 ID 号	
40135	当前用户坐标系号	显示 ID 号	
40136	当前系统模式	1:示教，2:自动，3:远程，4 无效	
40137	预留		
40138	实时显示机器人当前位置，根据点动坐标系类型来决定显示关节坐标还是世界坐标，每个坐标值都是 32 位浮点型数据，因此用两个寄存器表示一个坐标值	J1/X 坐标低位	机器人当前位置反馈，根据点动坐标系类型，来显示关节坐标或世界坐标。
40139		J1/X 坐标高位	
40140		J2/Y 坐标低位	
40141		J2/Y 坐标高位	
40142		J3/Z 坐标低位	
40143		J3/Z 坐标高位	
40144		J4/A 坐标低位	
40145		J4/A 坐标高位	
40146		J5/B 坐标低位	
40147		J5/B 坐标高位	
40148		J6/C 坐标低位	
40149		J6/C 坐标高位	
40150		坐标 mode 值	
40151		坐标 Cfg1	
40152		坐标 Cfg2	
40153		坐标 Cfg3	
40154		坐标 Cfg4	
40155		坐标 Cfg5	
40156		坐标 Cfg6	

40157	当前直接运动方式反馈	0:MovJ、1:MovL、2:MovArch、3:MovLArch	
40158	抬升高度反馈	参数低位	适用于 MovArch 与 MovLArch
40159		参数高位	
40160	下降高度反馈	参数低位	
40161		参数高位	
40162	相对过度 per 反馈	参数低位	
40163		参数高位	
40164	直接运动到位状态	0:未到位或无效，1:到位	
40165	点位变量 Px 读取是否成功	0:失败或无效，1:成功	
40166	点位变量 Px 写入是否成功	0:失败或无效，1:成功	
40167	点位数组变量 P[i]读取是否成功	0:失败或无效，1:成功	
40168	读取到的 Px 点位变量的值，x 的值在 40420 寄存器中设置，如当 40420 寄存器设为 1 时，触发读取命令，可读取到示教器上的 P1 变量	读取到的 J1/X 坐标低位	读取到的点位值
40169		读取到的 J1/X 坐标高位	
40170		读取到的 J2/Y 坐标低位	
40171		读取到的 J2/Y 坐标高位	
40172		读取到的 J3/Z 坐标低位	
40173		读取到的 J3/Z 坐标高位	
40174		读取到的 J4/A 坐标低位	
40175		读取到的 J4/A 坐标高位	
40176		读取到的 J5/B 坐标低位	
40177		读取到的 J5/B 坐标高位	
40178		读取到的 J6/C 坐标低位	
40179		读取到的 J6/C 坐标高位	
40180		读取到的坐标 mode 值	当点位类型为 CPOS 时有效
40181		读取到的坐标 Cfg1	
40182		读取到的坐标 Cfg2	
40183		读取到的坐标 Cfg3	
40184		读取到的坐标 Cfg4	
40185		读取到的坐标 Cfg5	
40186		读取到的坐标 Cfg6	
40187	读取到的 Px 变量的类型	0:APOS，1:CPOS	
40188	预留		
40189	预留		
40190	读取到的点位数组变量 P[i]的值，i 的值在 40421 寄存器	P[i]变量读取的 J1/X 坐标低位	i 的值会受到点位数组 count 的影响，默认取值范围为 1~1000。
40191		P[i]变量读取的 J1/X 坐标高位	
40192		P[i]变量读取的 J2/Y 坐标低位	

40193	中设置，如当40421 寄存器设为 1 时，触发读取命令，可读取到示教器上的 P[1]变量	P[i]变量读取的 J2/Y 坐标高位	
40194		P[i]变量读取的 J3/Z 坐标低位	
40195		P[i]变量读取的 J3/Z 坐标高位	
40196		P[i]变量读取的 J4/A 坐标低位	
40197		P[i]变量读取的 J4/A 坐标高位	
40198		P[i]变量读取的 J5/B 坐标低位	
40199		P[i]变量读取的 J5/B 坐标高位	
40200		P[i]变量读取的 J6/C 坐标低位	
40201		P[i]变量读取的 J6/C 坐标高位	
40202		坐标 mode 值	
40203		坐标 Cfg1	
40204		坐标 Cfg2	
40205		坐标 Cfg3	
40206		坐标 Cfg4	
40207		坐标 Cfg5	
40208		坐标 Cfg6	
40209	读取到的 P[i]变量的类型	0:APOS，1:CPOS	
40210	预留		
40211	预留		
40212	读取到的点位数组变量 P[i+n]的值，i 的值在 40421 寄存器中设置，n 的值在 40422 寄存器中设置，如当 40421 寄存器设为 1，40422 寄存器设为 4 时，触发读取命令，可读取到示教器上的 P[5]变量	P[i+n]变量读取的 J1/X 坐标低位	n 的取范围为 0~100，超出此范围，默认按 0 处理
40213		P[i+n]变量读取的 J1/X 坐标高位	
40214		P[i+n]变量读取的 J2/Y 坐标低位	
40215		P[i+n]变量读取的 J2/Y 坐标高位	
40216		P[i+n]变量读取的 J3/Z 坐标低位	
40217		P[i+n]变量读取的 J3/Z 坐标高位	
40218		P[i+n]变量读取的 J4/A 坐标低位	
40219		P[i+n]变量读取的 J4/A 坐标高位	
40220		P[i+n]变量读取的 J5/B 坐标低位	
40221		P[i+n]变量读取的 J5/B 坐标高位	
40222		P[i+n]变量读取的 J6/C 坐标低位	
40223		P[i+n]变量读取的 J6/C 坐标高位	
40224		坐标 mode 值	
40225		坐标 Cfg1	
40226		坐标 Cfg2	
40227		坐标 Cfg3	
40228		坐标 Cfg4	
40229		坐标 Cfg5	
40230		坐标 Cfg6	
40231	读取到的 P[i+n]变量的类型	0:APOS，1:CPOS	

40232	预留		
40233	预留		
40234~40250 为预留发送区域			
外部 PLC To Robot			
寄存器	**定义**	**说明**	**注释**
40251	Robot 操作指令	bit0：无	
		bit1：上使能/下使能	
		bit2：机器人程序启动	远程模式下生效
		bit3：机器人程序停止	
		bit4：机器人错误复位	
		bit5：切换到示教模式	
		bit6：切换到自动模式/远程模式	
		bit7：加载指定程序	远程模式下生效，需要先设置工程名及程序名,即在 40254~40263 中写入"工程名.程序名"的 ASCII 码值
		bit8：注销当前工程	远程模式下生效
		bit9：设置全局速度	使用 40253 的值进行设置
		bit10：指令状态机重置	命令执行错误时，将其置为 1 来进行复位
		bit11：写入当前 Px 变量	
		bit12：获取当前 Px 变量	
		bit13：复位程序指针至首行(不注销程序)	远程模式下生效
		bit14：重新加载远程自启动程序	远程模式下生效
		bit15：读取点位数组变量 P[i]以及 P[i+n]	
40252	心跳间隔时间	控制 40001.bit8 的心跳间隔	如果该值小于 500，一律按 500 处理
40253	全局速度值	待设置的全局速度值	取值范围 0~100
40254 ~ 40263	待加载的工程与程序名	待加载的"工程名.程序名"，以 ASCII 码值写入，共 10 个寄存器，最大支持 20 个字节	如 estun.main 对应的 16 进制数值分别为 0x7365、0x7574、0x2E6E、0x616D 和 0x6E69
40264	SimDI[1-16]	1~16 路虚拟 DI	显示虚拟 DI，寄存器的每个 bit 代表一路虚拟 DI(读写)
40265	SimDI[17-32]	17~32 路虚拟 DI	
40266	SimDI[33-48]	33~48 路虚拟 DI	
40267	SimDI[49-64]	49~64 路虚拟 DI	

40268	SimDI[65-80]	65~80 路虚拟 DI	
40269	SimDI[81-96]	81~96 路虚拟 DI	
40270	SimDI[97-112]	97~112 路虚拟 DI	
40271	SimDI[113-128]	113~128 路虚拟 DI	
40272 ~ 40303	SimAI[1-16]	1~16 路虚拟 AI	每两个寄存器对应一个模拟量，分别表示高低位，如 40272 表示 SimAI[1] 的低位，40273 表示 SimAI[1]的高位(读写)
40304 ~ 40335	SimAI[17-32]	17~32 路虚拟 AI	
40336 ~ 40351	SimAI[33-48]	33~48 路虚拟 AI	每一个寄存器对应一个模拟量，用于传输 INT 类型数据，如 40336 表示 SimAI[33](读写)
40352 ~ 40367	SimAI[49-64]	49~64 路虚拟 AI	
40368	命令权限寄存器	写入 0x11，打开 robot 命令下发权限	只有当此寄存器的值设为 0x11 时，才会响应命令寄存器的命令(40251，40382~40385 寄存器)
40369	点动模式选择	0：连续，1：寸动	
40370	点动 J1/X	往机器人正限位方向运动	寄存器值：0->1 启动，1->0 停止
40371	点动 J2/Y		
40372	点动 J3/Z		
40373	点动 J4/(四关节 C、六关节 A)		
40374	点动 J5 /六关节 B		
40375	点动 J6 /六关节 C		
40376	点动 J1/X	往机器人负限位方向运动	
40377	点动 J2/Y		
40378	点动 J3/Z		
40379	点动 J4/(四关节 C、六关节 A)		
40380	点动 J5/六关节 B		
40381	点动 J6/六关节 C		
40382	将机器人当前位置写入 Px 变量	当此寄存器值写入 1 时，触发示教点位命令，会将机器人当前位置写入点位变量 Px 中，x 的值在 40420 寄存器中设置	该寄存器为复归型，即外部写入 1，执行完成后，内部会复位成 0
40383	设置点动坐标系	当此寄存器写入指定值时，会设置点动坐标系	0:关节，1:世界，2:工具，3:用户

40384	设置工具坐标系 TOOLx	当此寄存器写入值时，会依据该值设置工具坐标系，如写入 1 时，会将当前工具坐标系设为 TOOL1(注：写入 0 时，会设置为 nullTool)	工具坐标系变量需提前在示教器上建好，否则会导致命令执行失败
40385	设置用户坐标系 USERCOORx	当此寄存器写入值时，会依据该值设置用户坐标系，如写入 1 时，会将当前用户坐标系设为 USERCOOR1(注：写入 0 时，会设置为 World)	用户坐标系变量需提前在示教器上建好，否则会导致命令执行失败
40386	直接运动到 Px 变量位置	x 的值在 40420 寄存器中设置	寄存器值：0->1 启动，1->0 停止
40387	设置当前直接运动方式	0:MovJ、1:MovL、2:MovArch、3:MovLArch	
40388	设置的上升高度	参数低位	适用于 MovArch 与 MovLArch
40389		参数高位	
40390	设置的上升高度	参数低位	
40391		参数高位	
40392	设置的相对过度 per	参数低位	
40393		参数高位	
40394	寸动关节步长	参数低位	取值范围 0.0~1.0 度，关节坐标下有效，超出此范围按 0.5 处理
40395		参数高位	
40396	寸动线性步长	参数低位	取值范围 0.0~10.0 毫米，直角坐标下有效，超出此范围按 1.0 处理
40397		参数高位	
40398	待写入 Px 点位变量的值，x 的值在 40420 寄存器中设置，如当 40420 寄存器设为 1 时，触发写入命令，可将寄存器中的值设置到示教器上的 P1 变量	待写入的 J1/X 坐标低位	待写入的点位值
40399		待写入的 J1/X 坐标高位	
40400		待写入的 J2/Y 坐标低位	
40401		待写入的 J2/Y 坐标高位	
40402		待写入的 J3/Z 坐标低位	
40403		待写入的 J3/Z 坐标高位	
40404		待写入的 J4/A 坐标低位	
40405		待写入的 J4/A 坐标高位	
40406		待写入的 J5/B 坐标低位	
40407		待写入的 J5/B 坐标高位	
40408		待写入的 J6/C 坐标低位	
40409		待写入的 J6/C 坐标高位	
40410		待写入的 mode 值	待写入的 confdata 值
40411		待写入的 Cfg1	
40412		待写入的 Cfg2	

40413		待写入的 Cfg3	
40414		待写入的 Cfg4	
40415		待写入的 Cfg5	
40416		待写入的 Cfg6	
40417	待写入 Px 点位变量的类型	0：APOS，1：CPOS	在写入 CPOS 类型点位时，必须给该寄存器赋值为 1，否则会将 CPOS 刷成 APOS
40418	预留		
40419	预留		
40420	Px 变量名的序号	用于读、写、运动等操作	如写入 1.2.3…就表示变量 P1.P2.P3…
40421	点位数组 P 变量的序号 i(1~1000)	用于读取点位数组 P 中的元素	P[i]
40422	点位数组 P 变量序号的偏移 n(0-100)	用于读取点位数组 P 中的元素	P[i+n]
40423~40500 为预留接收区域			
40501~41500 为用户自定义区域			