

DUCO CORE

对外通讯接口说明

文档版本 V1.5

适用软件版本 V2.3.0 及以上

中科新松有限公司

2022 年 07 月 29 日

版权所有©中科新松有限公司 2022。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制文档内容的部分或全部，并不得以任何形式传播。

商标声明

“SIASUN”、“新松”、“SIASUN 新松”等文字或形象均进行了商标注册保护，注册商标信息可见于公开的商标注册信息中。

本手册提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受中科新松有限公司商业合同和条款的约束，本手册中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围内。除非合同另有约定，中科新松有限公司对本手册内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本手册内容会不定期进行更新。除非另有约定，本手册仅作为使用指导，本手册中的所有陈述、信息和建议不构成任何明示或暗示的担保。

中科新松有限公司

地址：上海市浦东新区金藏路 257 号 邮编：201206

网址：<http://www.siasun-in.com>

修订历史

1. 2021 年 6 月 21 日 文档版本号 V1.0 软件版本 V1.0.2 首次修订
2. 2021 年 7 月 31 日 文档版本号 V1.1 软件版本 V1.1.0 优化描述
3. 2021 年 11 月 20 日 文档版本号 V1.2 软件版本 V1.3.0 添加额外控制端口数据
 - 1) 2001 端口增加数据。
 - 2) Modbus 增加数据。
4. 2022 年 1 月 11 日 文档版本号 V1.3 软件版本 V1.4.0 添加配方接口
5. 2022 年 7 月 7 日 文档版本号 V1.4 软件版本 V2.2.1 添加程序状态机
6. 2022 年 7 月 29 日 文档版本号 V1.5 软件版本 V2.3.0 第 5.1 章节 ForceEnd 修改成末端 TCP 下的力描述信息

目录

修订历史	4
1 前言	6
1.1 读者对象	6
1.2 符号约定	6
1.3 特别声明	7
2 安全	8
3 TCP/IP 接口	9
3.1 2000 端口	9
3.2 2001 端口	11
4 MODBUS TCP SERVER	17
5 配方接口	20
5.1 系统数据映射说明	20
5.2 配方的创建	22
5.2.1 流式数据配方	22
5.2.2 CAN 协议配方	25
5.3 配方的使用	28
5.3.1 485 端口配方的使用	28
5.3.2 CAN 端口配方的使用	30
5.3.3 TCP/UDP 端口配方的使用	33

1 前言

1.1 读者对象

本文档提供新松协作机器人使用说明，便于用户详细了解机器人基本信息，更安全方便地使用机器人。请务必在认真阅读并充分理解本文档的基础上操作本机器人。






本文档主要适用于以下用户：

- 机器人现场工程师
- 机器人软件工程师
- 硬件安装工程师
- 现场维护工程师
- 系统维护工程师

经过基本培训的操作人员才允许进行该机器人的操作。

1.2 符号约定

在本文中可能出现下列标志，它们所代表的含义如下：

符号	说明
 危险	用于警示紧急的危险情形，若不加避免，将会导致人员死亡或严重的人身伤害。
 警告	用于警示潜在的危险情形，若不加避免，可能会导致人员死亡或严重的人身伤害。
 小心	用于警示潜在的危险情形，若不加避免，可能会导致中度或轻微的人身伤害。
 注意	用于传递设备或环境安全警示信息，若不加避免，可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。 “注意”不涉及人身伤害。
 提示	用于突出重要/关键信息、最佳实践和小窍门等。 “提示”不是安全警示信息，不涉及人身、设备及环境伤害信息。

1.3 特别声明

本手册仅作为使用指导，其内容（如设备外观、软件界面）依据实验室设备信息编写。手册提供的内容具有一般性的指导意义，并不确保涵盖所有型号产品的所有使用场景。因版本升级、设备型号不一致等原因，可能造成手册中提供的内容与用户使用的设备不一致。请以用户设备的信息为准，本手册不再针对前述情况造成的差异一一说明。

本手册中提供的最大值是设备在满足相应标准的实验室特定场景（例如，恒温恒湿无干扰环境、典型工况等）达到的最大值。在实际工况中，由于工作环境、具体工况、测试方法不一致等原因会使设备测试出的最大值与手册中提供的数据不一致。

2 安全

关于机器人安全使用说明，请参照 DUCO CORE 用户手册。

3 TCP/IP 接口

机器人在上电状态下会作为 TCP Server 监听端口：2000 和 2001 的连接。端口 2000 可接收对机器人的控制命令及返回状态。端口 2001 会以 10Hz 的频率向外部发送机器人当前的状态信息。

3.1 2000 端口

2000 端口采用请求-响应式的应答机制，可以接受系统指令。

Client 和 Server 间的通讯协议如下：

接收的指令	返回值	说明
run(程序名) 或 run(程序名, 速度百分比)	开始时反馈: run start 执行失败反馈: run fail	机器人会执行相应程序, 不设置速度百分比或者速度百分比参数无效时, 使用当前速度运行。
speed(速度百分比)	执行成功反馈: set success 执行失败反馈: set fail	可修改程序运行的速度百分比。参数范围在 (0, 100], 若修改成功, 会返回 set success, 若不成功, 会返回 set fail
stop	执行成功反馈: stop success 执行失败反馈: stop fail	机器人会停止当前执行的指令。停止后会发送 stop success 回执。若机器人处于非运行状态, 接收到 stop 命令后会发送 stop fail 回执
pause	执行成功反馈: pause success 执行失败反馈: pause fail	机器人会暂停当前执行的指令。停止时会发送 pause success 回执。若机器人处于非运行状态, 接收到 pause 命令后会发送 pause fail 回执。
resume	执行成功反馈: resume success 执行失败反馈: resume fail	机器人在暂停时会继续运行程序。运行时会发送 resume success 回执。若机器人不在暂停状态, 接收到 resume 命令后会发送 resume fail 回执
poweron	执行成功反馈: poweron success 执行失败反馈: poweron fail	机器人上电, 若上电成功, 建立通讯链接, 返回 poweron success 若失败返回 poweron fail

接收的指令	返回值	说明
poweroff	执行成功反馈: poweroff success 执行失败反馈: poweroff fail	机器人下电, 若下电成功, 返回 poweroff success. 若失败返回 poweroff fail
enable	执行成功反馈: enable success 执行失败反馈: enable fail	机器人上使能, 若上使能成功, 返回 enable success 若失败返回 enable fail
disable	执行成功反馈: disable success 执行失败反馈: disable fail	机器人下使能, 若下使能成功, 返回 disable success 若失败返回 disable fail
shutdown	执行成功反馈: shutdown success 执行失败反馈: shutdown fail	机器人已处于 poweroff 状态, 机器人系统会自动关闭, 返回 shutdown success 回执。若机器人未下电, 返回 shutdown fail 回执
state	反馈各系统状态机, 格式: (1): (2): (3): (4) 注: (1) 机器人状态机, (2) 机器人程序运行状态, (3) 安全监控状态, (4) 机器人操作模式	各状态机以数字形式返回。每个数字所代表的状态。请参考 2001 端口, 地址 1448-1451
isprogfinish	反馈程序是否运行结束, 结束反馈字符 1, 未结束反馈字符 0	
getlasterror	反馈机器人最后一次报错的详细错误信息, 格式 error_id:error_message\n	错误码返回为 16 进制

3.2 2001 端口

端口 2001 会以 10Hz 的频率向外部发送机器人当前的状态信息，数据总长度 1468 个字节。

数据	类型	数量	字节大小	地址	注释
关节实际位置	float	7	28	0-27	6 关节各自实际位置，第 7 位预留。 单位为 rad
关节实际速度	float	7	28	28-55	6 关节各自实际速度，第 7 位预留。 单位为 rad/s
关节实际加速度	float	7	28	56-83	6 关节各自实际加速度，第 7 位预留。 单位为 rad/s^2
关节实际力矩	float	7	28	84-111	6 关节各自实际力矩，第 7 位预留。 单位为 Nm
关节期望位置	float	7	28	112-139	6 关节各自设定位置，第 7 位预留。 单位为 rad
关节期望速度	float	7	28	140-167	6 关节各自设定速度，第 7 位预留。 单位为 rad/s
关节期望加速度	float	7	28	168-195	6 关节各自设定加速度，第 7 位预留。 单位为 rad/s^2
关节期望力矩	float	7	28	196-223	6 关节各自设定力矩，第 7 位预留。 单位为 Nm
关节实际温度	float	7	28	224-251	7 关节各自实际温度。数据预留
关节实际电流	float	7	28	252-279	6 关节各自实际电流，第 7 位预留。 单位为额定电流千分比。
伺服驱动错误 id	uint	7	28	280-307	6 关节伺服驱动错误 id，第 7 位预留。
伺服驱动状态字	uint	7	28	308-335	6 关节伺服驱动状态字，第 7 位预留
预留	byte	32	32	336-367	

TCP 实际位置	float	6	24	368-391	末端笛卡尔空间实际位姿。TCP 相对基坐标系的值。数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 m, rad
TCP 实际速度	float	6	24	392-415	末端笛卡尔空间实际速度。TCP 相对基坐标系的值。数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 m/s, rad/s
TCP 实际加速度	float	6	24	416-439	末端笛卡尔空间实际速度。TCP 相对基坐标系的值。数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 m/s^2 , rad/s^2
法兰实际外力	float	6	24	440-463	机器人末端法兰受到的外力, 当配置末端力矩传感器时, 为传感器数据在机器人末端法兰面下的力描述。数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 N, Nm
TCP 期望位置	float	6	24	464-487	末端笛卡尔空间实际位姿。TCP 相对基坐标系的值。数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 m, rad
TCP 期望速度	float	6	24	488-511	末端笛卡尔空间实际速度。TCP 相对基坐标系的值。数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 m/s, rad/s
TCP 期望加速度	float	6	24	512-535	末端笛卡尔空间实际速度。TCP 相对基坐标系的值。数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 m/s^2 , rad/s^2
法兰理论外力	float	6	24	536-559	机器人末端法兰在当前运动状态下, 理论上收到的外力大小。数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 N, Nm
基座实际外力	float	6	24	560-583	底座实际受力在机器人基座下的力描述。当配置底座力矩传感器时, 该值为力矩传感器测量值, 数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 N, Nm
基座理论外力	float	6	24	584-607	在当前运动状态下, 底座理论受力在机器人基座下的力描述。数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 N, Nm

当前激活的工具坐标系	float	6	24	608-631	数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 m, rad
当前激活的工件坐标系	float	6	24	632-655	不包含当前工具坐标系偏移量. 数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 m, rad
预留	byte	64	64	656-719	
功能数字 IO 输入	byte	8	8	720-727	以一个 byte 表示一个 bool 量。True=1, false=0。对应硬件 FDI1-FDI8
功能数字 IO 输出	byte	8	8	728-735	以一个 byte 表示一个 bool 量。True=1, false=0。对应硬件 FD01-FD08
数字 IO 输入	byte	16	16	736-751	对应 DI 接口普通 DI 输入 1-16。对应硬件 DI1-DI16
数字 IO 输出	byte	16	16	752-767	对应 DO 接口普通 DO 输出 1-16。对应硬件 D01-D016
模拟量输入	float	8	32	768-799	8 个模拟量输入, 前 4 路为电流、后 4 路为电压。对应硬件 AI_C1-AI_C4, AI_V1-AI_V4。取值范围详见硬件说明书。
模拟量输出	float	8	32	800-831	预留
float 寄存器 输入	float	32	128	832-959	可通过 modbus、Profinet 等外部接口修改
float 寄存器 输出	float	32	128	960-1087	可通过脚本函数修改
功能 bool 寄存器 输入	byte	16	16	1088-1103	可通过 modbus、Profinet 等外部接口修改
功能 bool 寄存器 输出	byte	16	16	1104-1119	可通过脚本函数修改
bool 寄存器 输入	byte	64	64	1120-1183	可通过 modbus、Profinet 等外部接口修改

bool 寄存器输出	byte	64	64	1184-1247	可通过脚本函数修改
word 寄存器输入	char	32;	64	1248-1311	可通过 modbus、Profinet 等外部接口修改
word 寄存器输出	char	32	64	1312-1375	可通过脚本函数修改
预留	byte	32	32	1376-14047	
工具 I0 输入	byte	8	8	1408-1415	工具数字输入（1-2 末端输入，3-8 预留）。对应硬件工具 I0 数字输入 1-2
工具 I0 输出	byte	8	8	1416-1423	工具数字输出（1-2 末端输出，3-8 预留）。对应硬件工具 I0 数字输出 1-2
工具模拟量输入	float	2	16	1424-1431	工具末端电压模拟量输入。取值范围 0V-10V
工具模拟量输出	float	2	16	1432-1439	预留
工具按钮状态	byte	2	2	1440-1441	以一个 byte 表示一个 bool 量。True=1, false=0。地址 1440 表示 S 键的状态，地址 1441 表示 T 键的状态。
预留	byte	6	6	1442-1447	
机器人操作模式	char	1	1	1448	0: kManual, 1: kAuto, 2: kRemote,
机器人状态	char	1	1	1449	0: SR_Start, 1: SR_Initialize, 2: SR_Logout, 3: SR_Login, 4: SR_PowerOff, 5: SR_Disable/SR_PowerOn（上电未使能状态） 6: SR_Enable

机器人程序运行状态	char	1	1	1450	0: SP_Stopped, 1: SP_Stopping, 2: SP_Running, 3: SP_Paused, 4: SP_Pausing 5: SP_TaskRunning
安全监控状态	char	1	1	1451	0: SS_INIT 2: SS_WAIT 3: SS_CONFIG 4: SS_POWER_OFF 5: SS_RUN 6: SS_RECOVERY 7: SS_STOP2 8: SS_STOP1 9: SS_STOP0 10: SS_MODEL 12: SS_REDUCE 13: SS_BOOT 14: SS_FAIL 15: SS_UPDATE
碰撞检测触发信号	char	1	1	1452	为 1 表示触发碰撞检测
碰撞轴	char	1	1	1453	返回值为数字。1~6 表示发生碰撞的关节, 11~16 表示基于底座力传感器碰撞发生的方向。顺序为 X , Y, Z, Rx, Ry, Rz . 20 表示基于末端力传感器碰撞发生
预留	byte	2	2	1454-1455	
机器人错误代码	uint	1	4	1456-1459	机器人最新报错的错误代码, 当同时出现多个错误时, 可能返回其中的任何一个错误代码。

预留	byte	8	8	1460-1467	
----	------	---	---	-----------	--

4 Modbus TCP Server

本章描述机器人端作为 Modbus/TCP server 端口时的可交互数据。

线圈 (Colis)：读 (功能码：1) 写 (功能码：5 单个/15 多个)

Addr	节点说明	数量	读写权限	功能
0-15	功能 bool 寄存器输入	16	R/W	外部修改寄存器，触发功能
16-31	功能 bool 寄存器输出	16	R	寄存器值改变，表示对应系统状态
32-95	bool 寄存器 输入	64	R/W	外部修改寄存器，在脚本中读取
96-159	bool 寄存器 输出	64	R	脚本或 UI 中修改寄存器，输出到外部
160-167	功能数字 IO 输入	8	R	外部物理 IO 输入，触发功能
168-175	功能数字 IO 输出	8	R	物理 IO 输出，表示对应系统状态
176-191	数字 IO 输入	16	R	外部物理 IO 输入，在（脚本/UI/外部端口）中读取
192-207	数字 IO 输出	16	R/W	（脚本/UI/外部端口）中修改 IO，输出到外部物理 IO
208-215	工具 IO 输入	8	R	外部物理 IO 输入，在（脚本/UI/外部端口）中读取
216-223	工具 IO 输出	8	R/W	（脚本/UI/外部端口）中修改 IO，输出到外部物理 IO
224-231	工具末端按钮	8	R	工具末端按钮输入，触发功能

Holding Registers:读 (功能码：3) 写 (功能码：6 单个/16 多个)

Addr	节点说明	读写权限
0	机器人操作模式 0: kManual 1: kAuto 2: kRemote	R

Addr	节点说明	读写权限
1	机器人状态 0: SR_Start 1: SR_Initialize 2: SR_Logout 3: SR_Login 4: SR_PowerOff 5: SR_Disable/SR_PowerOn（上电未使能状态） 6: SR_Enable	R
2	程序状态 0: SP_Stopped 1: SP_Stopping 2: SP_Running 3: SP_Paused 4: SP_Pausing 5: SP_TaskRunning	R
3	机器人安全状态 0: SS_INIT 2: SS_WAIT 3: SS_CONFIG 4: SS_POWER_OFF 5: SS_RUN 6: SS_RECOVERY 7: SS_STOP2 8: SS_STOP1 9: SS_STOP0 10: SS_MODEL 12: SS_REDUCE 13: SS_BOOT 14: SS_FAIL 15: SS_UPDATE	R
4-7	保留	R
8-39	World 输入寄存器	R/W

Addr	节点说明	读写权限
40-71	World 输出寄存器	R
72-79	控制柜模拟输入, 采用 count 表示, 实际电流值为 count/100, 电压值为 count/1000。	R
80-87	控制柜模拟输出。(预留)	R/W
88-89	末端模拟输入, 采用 count 表示, 实际电流值为 count/100, 电压值为 count/1000。	R
90-91	末端模拟输出。(预留)	R/W
92-98	关节实际位置, 取值-32767-32767, 单位 1/1000rad。	R
99-105	关节实际速度, 取值-32767-32767, 单位 1/1000rad/s。	R
106-111	空间实际位置, 取值-32767-32767, 单位 1/10000m; 1/10000 rad	R
112-117	空间实际速度, 取值-32767-32767, 单位 1/10000 (m/s) ; 1/10000 (rad/s)	R

5 配方接口

配方分类：

- (1) 根据功能，分为用户自定义配方（非实时配方）和系统数据配方（实时配方）：
自定义配方：由用户自己定义想要接收和发送的数据格式，由配方管理器提取用户的输入有效数据和对输出数据进行组帧。接收和发送的过程均由用户在脚本程序中控制。
系统数据配方：用户从系统提供的系统数据中选择想要的数据通过输出端进行实时读取，或者通过输入端实时修改机器人提供的系统变量，用户可配置的变量均由系统提供，自用无法自定义，数据的接收法发送不受用户控制，由系统实时控制，控制周期（发送和接收的最快周期）为 4ms。
- (2) 根据协议，分为流式协议配方和 can 协议配方：
流式协议配方为流式数据（485、tcp 等）服务。
can 协议配方仅提供对 can 的支持。

5.1 系统数据映射说明

配方的数据类型：

- (1) byte(无符号 8 位整形)
- (2) char(有符号 8 位整形)
- (3) word(无符号 16 位整形)
- (4) short(有符号 16 位整形)
- (5) dword(无符号 32 位整形)
- (6) int(有符号 32 位整形)
- (7) float(32 位浮点数)
- (8) double(64 位双精度浮点数)

系统数据输入：

数据方向：用户→机器人

配方变量	类型	对应的接口	说明
DigitalOutputCommand[1..2]	byte	控制柜通用输出 DO[1..16]	用户控制机器人的 DO。一个 byte 顺序对应 8 个 DO。
DigitalOutputCommand[3]	byte	机械臂末端工具输出 Tool_DO[1..2]	byte 的前两位对应工具 DO1 和 DO2，后六位预留。
BitInputReg[1..2]	byte	寄存器功能输入 fun_reg_in[1..16]	一个 byte 顺序对应 8 个寄存器功能输入
BitInputReg[3..10]	byte	寄存器 Bool 输入 bool_reg_in[1..64]	一个 byte 顺序对应 8 个寄存器 Bool 输入
WordInputReg[1..32]	word	寄存器 Word 输入 word_reg_in[1..32]	按顺序对应寄存器 Word 输入 1-32
FloatInputReg[1..32]	float	寄存器 Float 输入 float_reg_in[1..32]	按顺序对应寄存器 Float 输入 1-32

ForceEnd[1..6]	float	末端力传感器数据	末端力传感器数据在机器人末端 TCP 下的力描述。数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 N, Nm
ForceBase[1..6]	float	基座力传感器数据	底座力传感器数据在机器人基座下的力描述。数据定义: X, Y, Z, Rx, Ry, Rz, 单位为 N, Nm

系统数据输出:

数据方向: 机器人→用户

配方变量	类型	对应的接口	说明
DigitalSignalStatus[1..2]	byte	控制柜通用输入 DI[1..16]	机器人向用户反馈当前机器人的输入信号。一个 byte 顺序对应 8 个 DI
DigitalSignalStatus[3]	byte	机械臂末端工具输入 Tool_DI[1..2]	byte 的前两位对应工具 DI1 和 DI2, 后六位预留
DigitalSignalStatus[4..5]	byte	控制柜通用输出 DO[1..16]	机器人向用户反馈当前机器人的输出信号。一个 byte 顺序对应 8 个 DO
DigitalSignalStatus[6]	byte	工具输出 Tool_DO [1..2]	byte 的前两位对应工具 DO1 和 DO2, 后六位预留
DigitalSignalStatus[7]	byte	IO 功能输入 fun_io_in[1..8]	一个 byte 顺序对应功能输入的 1-8
DigitalSignalStatus[8]	byte	IO 功能输出 fun_io_out [1..8]	一个 byte 顺序对应功能输出的 1-8
RobotState[1..7]	float	关节实际位置	6 关节各自实际位置, 第 7 位预留。单位为 rad
RobotState[8..13]	float	末端在基坐标系下的实际笛卡尔位姿	末端笛卡尔空间实际位姿。TCP 相对基坐标系的值。数据定义: X,Y,Z,Rx,Ry,Rz, 单位为 m, rad
RobotState[14..19]	float	笛卡尔实际力矩	末端力传感器数据在机器人末端法兰面下的力描述。数据定义: X,Y,Z,Rx,Ry,Rz, 单位为 N, Nm
RobotState[20..25]	float	工具坐标系偏移量	工具坐标系下的位姿偏移量
RobotState[26..29]	float	负载质量和质心	末端负载的质量和质心, 单位 kg,m
RobotState[30..32]	float	预留	无
BitOutputReg[1..2]	byte	寄存器功能输出 fun_reg_out[1..16]	一个 byte 顺序对应 8 个寄存器功能输出
BitOutputReg[3..10]	byte	寄存器 Bool 输出	一个 byte 顺序对应 8 个寄存器

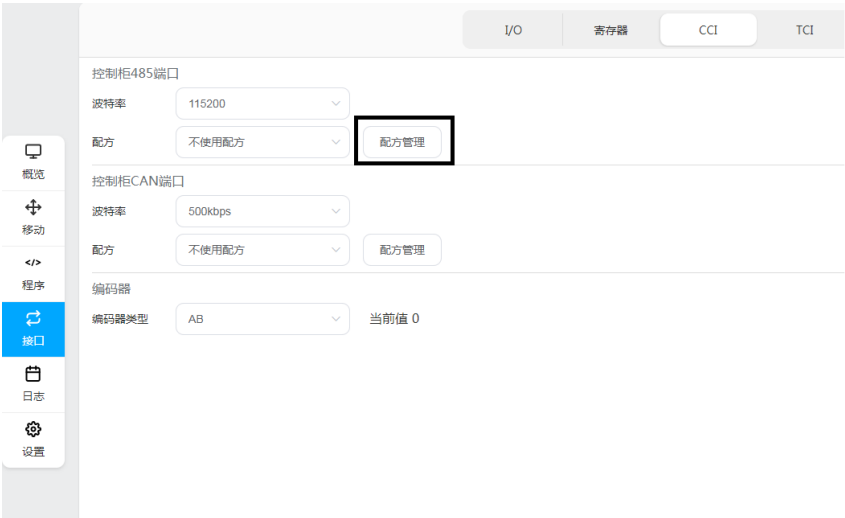
		bool_reg_out[1..64]	Bool 输出
WordOutputReg[1..32]	word	寄存器 Word 输出 word_reg_out[1..32]	按顺序对应寄存器 Word 输入 1-32
FloatOutputReg[1..32]	float	寄存器 Float 输出 float_reg_out[1..32]	按顺序对应寄存器 Float 输入 1-32

5.2 配方的创建

5.2.1 流式数据配方

5.2.1.1 用户自定义配方

1. 在 ui 界面选择任意流式端口（485、tcp 等）配方管理（这里选择 485 端口）



2. 点击添加，输入配方名，选择非实时配方



3. 添加输入数据项

定义数据项

输入

输出

添加数据项

帧头

255

255

+

-

帧尾

238

238

+

-

input1

byte

-

input2

byte

-

input3

byte

-

取消

确定

4. 添加输出数据项

定义数据项

输入

输出

添加数据项

帧头

255

255

+

-

帧尾

254

254

+

-

output1

byte

-

output2

byte

-

output3

byte

-

取消

确定

点击确认，配方创建完成。注意：输入与输出的帧头帧尾，是相对独立的，需单独配置，互不影响。

5.2.1.2 系统数据配方

- 1. 在ui界面选择任意流式端口（485、tcp等）配方管理（这里选择485端口）
- 2. 点击添加，输入配方名，选择非实时配方



3. 添加输入数据项



4. 添加输出数据项

定义数据项

输入 输出 添加数据项

帧头	255	255	+	-
帧尾	254	254	+	-
__RobotState_8	float	-		
__RobotState_9	float	-		
__RobotState_10	float	-		
__RobotState_11	float	-		

取消 确定

5. 点击确认，配方创建完成。

5.2.2 CAN 协议配方

5.2.2.1 用户自定义配方

1. 在ui界面选择can端口配方管理

I/O 寄存器 CCI

控制柜485端口

波特率 115200

配方 不使用配方 配方管理

控制柜CAN端口

波特率 500kbps

配方 不使用配方 配方管理

编码器

编码器类型 AB 当前值 0

2. 点击添加，输入配方名，选择非实时配方



3. 添加输入数据项



4. 添加输出数据项



点击确认，配方创建完成。

5.2.2.2 系统数据配方

- 1. 在ui界面选择can端口配方管理
- 2. 点击添加，输入配方名，选择实时配方
- 3. 添加输入数据项

定义数据项 ×

输入 输出 添加ID

ID: 1 添加数据项 删除

__DigitalOutputCommand_1

byte —

取消 确定

- 4. 添加输出数据项

定义数据项 ×

输入 输出 添加ID

ID: 2 添加数据项 删除

__RobotState_8

float —

__RobotState_9

float —

ID: 3 添加数据项 删除

__RobotState_10

float —

__RobotState_11

float —

ID: 4 添加数据项 删除

取消 确定

- 5. 点击确认，配方创建完成。

5.3 配方的使用

5.3.1 485 端口配方的使用

5.3.1.1 自定义配方使用

- 1. 使用485转串口调试助手将电脑和机器人485端口连接
- 2. 为485端口选择配方文件

控制柜485端口

波特率

115200

▼

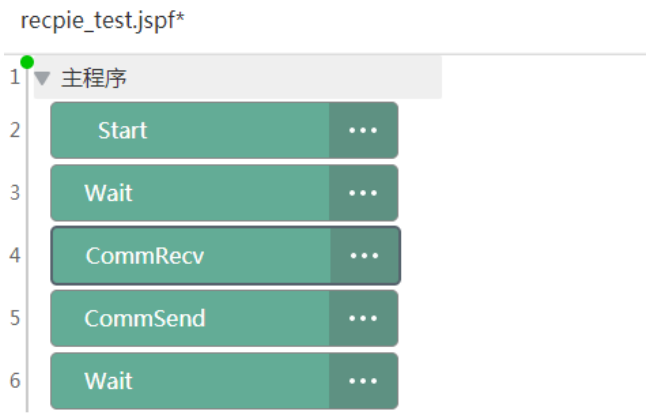
配方

custom

▼

配方管理

- 3. 编写程序流程控制



CommRecv:

接收数据

通讯端口

控制柜485端口

▼

接收数据

list1

▼

配方表

input1	byte
input2	byte
input3	byte

添加注释

添加注释

CommSend:

发送数据

通讯端口

控制柜485端口

返回变量

选择变量

配方表

output1	byte	125
output2	byte	125
output3	byte	125

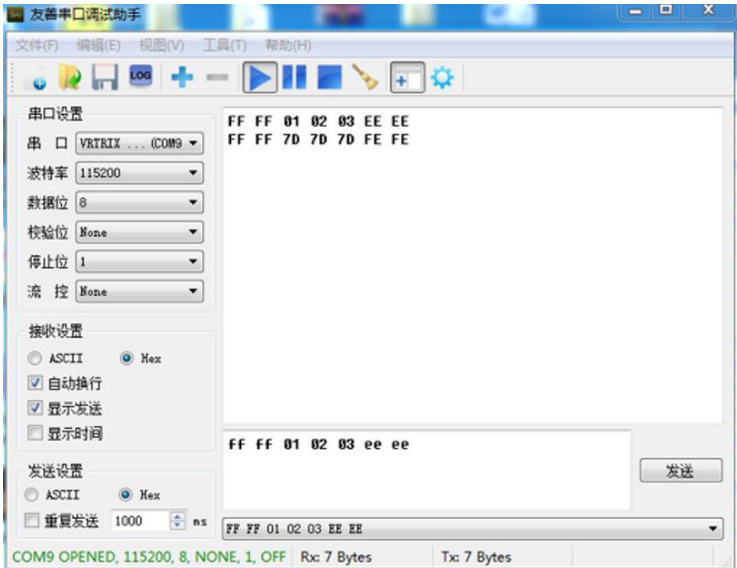
添加注释

添加注释

4. 运行程序，串口调试助手发送FF FF 01 02 03 EE EE
5. 程序变量界面显示

程序变量		
名称	类型	当前值
list1	num_list	{1,2,3}

6. 在串口调试助手端将会收到：FF FF 7D 7D 7D FE FE



5.3.1.2 系统配方使用

1. 使用485转串口调试助手将电脑和机器人485端口连接。
2. 为485端口选择配方文件

控制柜485端口

波特率

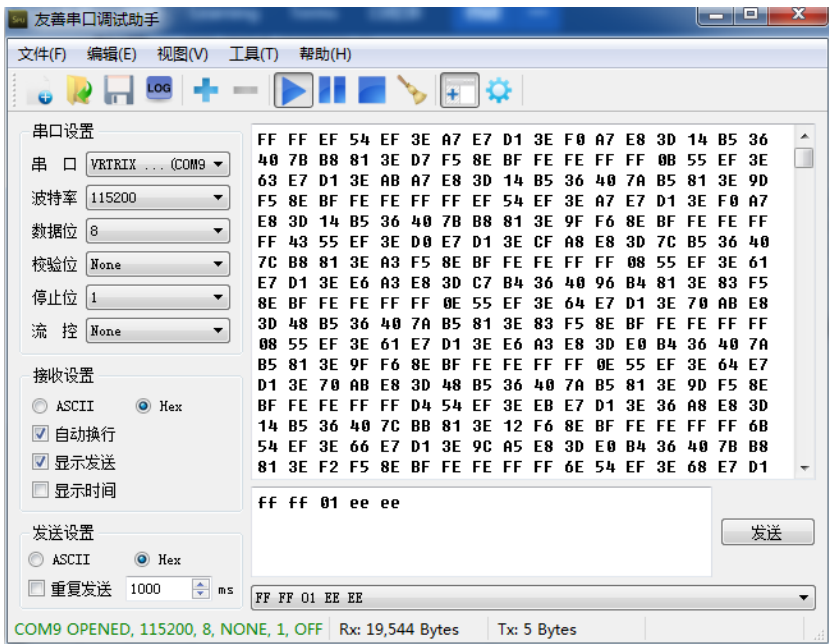
115200

配方

system

配方管理

3. 由于选择的是实时系统配方，串口调试助手将会周期性的收到机器人位姿数据：



4. 当通过串口调试助手发送FF FF 01 EE EE时，通过配方管理器将会控制对应的IO端口

功能输入 功能输出 通用输入 通用输出 模拟量					
类型	名称	说明	Modbus地址	状态	
通用输出 1	DO1		192	<div></div>	
通用输出 2	DO2		193	<div></div>	
通用输出 3	DO3		194	<div></div>	
通用输出 4	DO4		195	<div></div>	
通用输出 5	DO5		196	<div></div>	
通用输出 6	DO6		197	<div></div>	
通用输出 7	DO7		198	<div></div>	

5.3.2 CAN 端口配方的使用

5.3.2.1 自定义配方使用

- 1 使用can盒将电脑和机器人连接，使用CanTest调试can
- 2 为can端口选择配方文件

控制柜CAN端口

波特率

500kbps

配方

custom_can

配方管理

3 编写程序流程控制



CommRecv:

添加

变量

参数

运行

接收数据

通讯端口

控制柜CAN端口

接收数据

list1

配方表

ID: 1

input1

byte

input2

byte

input3

byte

添加注释

添加注释

CommSend:

添加

变量

参数

运行

发送数据

通讯端口

控制柜CAN端口

返回变量

选择变量

选择帧ID

2

配方表

output1

byte

125

output2

byte

125

output3

byte

125

添加注释

添加注释

4 运行程序，CanTest发送数据帧id 1, 数据01 02 03

5 程序变量界面显示:

程序变量		
名称	类型	当前值
list1	num_list	{1,1,2,3}

列表第一个变量为收到的数据帧id，后面的为数据。

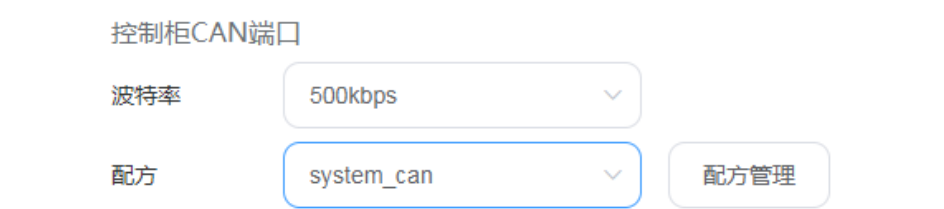
6 在CanTest端将会收到帧id为1: 7D 7D 7D(125 125 125)



序号	传输方向	时间标识	帧ID	帧格式	帧类型	数据长度	数据(HEX)
00000000	发送	14:33:14.2...	0x00000001	数据帧	标准帧	0x03	01 02 03
00000001	接收	14:33:17.2...	0x00000002	数据帧	标准帧	0x03	7d 7d 7d

5.3.2.2 系统配方使用

- 1. 使用can盒将电脑和机器人连接，使用CanTest调试can
- 2. 为can端口选择配方文件



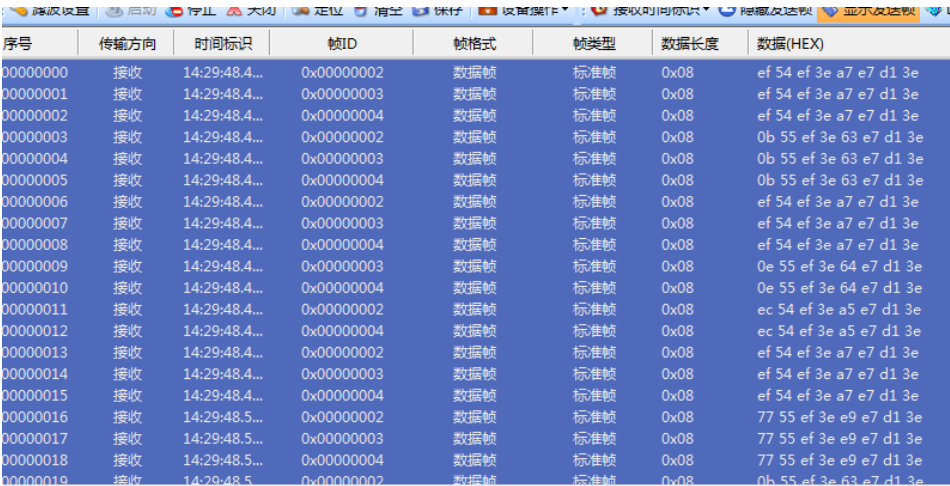
控制柜CAN端口

波特率: 500kbps

配方: system_can

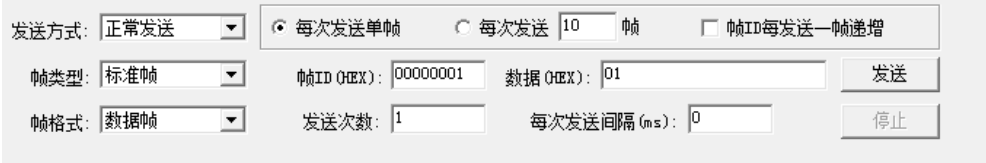
配方管理

由于选择的是实时系统配方，CanTest将会周期性的收到机器人位姿数据：



序号	传输方向	时间标识	帧ID	帧格式	帧类型	数据长度	数据(HEX)
00000000	接收	14:29:48.4...	0x00000002	数据帧	标准帧	0x08	ef 54 ef 3e a7 e7 d1 3e
00000001	接收	14:29:48.4...	0x00000003	数据帧	标准帧	0x08	ef 54 ef 3e a7 e7 d1 3e
00000002	接收	14:29:48.4...	0x00000004	数据帧	标准帧	0x08	ef 54 ef 3e a7 e7 d1 3e
00000003	接收	14:29:48.4...	0x00000002	数据帧	标准帧	0x08	0b 55 ef 3e 63 e7 d1 3e
00000004	接收	14:29:48.4...	0x00000003	数据帧	标准帧	0x08	0b 55 ef 3e 63 e7 d1 3e
00000005	接收	14:29:48.4...	0x00000004	数据帧	标准帧	0x08	0b 55 ef 3e 63 e7 d1 3e
00000006	接收	14:29:48.4...	0x00000002	数据帧	标准帧	0x08	ef 54 ef 3e a7 e7 d1 3e
00000007	接收	14:29:48.4...	0x00000003	数据帧	标准帧	0x08	ef 54 ef 3e a7 e7 d1 3e
00000008	接收	14:29:48.4...	0x00000004	数据帧	标准帧	0x08	ef 54 ef 3e a7 e7 d1 3e
00000009	接收	14:29:48.4...	0x00000003	数据帧	标准帧	0x08	0e 55 ef 3e 64 e7 d1 3e
00000010	接收	14:29:48.4...	0x00000004	数据帧	标准帧	0x08	0e 55 ef 3e 64 e7 d1 3e
00000011	接收	14:29:48.4...	0x00000002	数据帧	标准帧	0x08	ec 54 ef 3e a5 e7 d1 3e
00000012	接收	14:29:48.4...	0x00000004	数据帧	标准帧	0x08	ec 54 ef 3e a5 e7 d1 3e
00000013	接收	14:29:48.4...	0x00000002	数据帧	标准帧	0x08	ef 54 ef 3e a7 e7 d1 3e
00000014	接收	14:29:48.4...	0x00000003	数据帧	标准帧	0x08	ef 54 ef 3e a7 e7 d1 3e
00000015	接收	14:29:48.4...	0x00000004	数据帧	标准帧	0x08	ef 54 ef 3e a7 e7 d1 3e
00000016	接收	14:29:48.5...	0x00000002	数据帧	标准帧	0x08	77 55 ef 3e e9 e7 d1 3e
00000017	接收	14:29:48.5...	0x00000003	数据帧	标准帧	0x08	77 55 ef 3e e9 e7 d1 3e
00000018	接收	14:29:48.5...	0x00000004	数据帧	标准帧	0x08	77 55 ef 3e e9 e7 d1 3e
00000019	接收	14:29:48.5...	0x00000002	数据帧	标准帧	0x08	0b 55 ef 3e 63 e7 d1 3e

当通过CanTest发送数据帧id 1，数据为01时，通过配方管理器将会控制对应的IO端口：



发送方式: 正常发送

☒ 每次发送单帧 ☐ 每次发送 10 帧 ☐ 帧ID每发送一帧递增

帧类型: 标准帧 帧ID (HEX): 00000001 数据 (HEX): 01 发送

帧格式: 数据帧 发送次数: 1 每次发送间隔 (ms): 0 停止

<div>功能输入 功能输出 通用输入 通用输出 模拟量</div>				
类型	名称	说明	Modbus地址	状态
通用输出 1	DO1		192	<div></div>
通用输出 2	DO2		193	<div></div>
通用输出 3	DO3		194	<div></div>
通用输出 4	DO4		195	<div></div>
通用输出 5	DO5		196	<div></div>
通用输出 6	DO6		197	<div></div>
通用输出 7	DO7		198	<div></div>

5.3.3 TCP/UDP 端口配方的使用

tcp/ip 接口同样可以使用流式配方接口，由于历史原因，tcp 和 udp 仅仅能够使用配方中的系统数据配方（实时配方）。

机器人控制器提供了两种 tcp/ip 接口使用配方的方式：

- 1. udp server，端口号 2011。
- 2. tcp/udp client，ip 和端口由用户自定义。

5.3.3.1 udp server

- 1. 使用网线将电脑和控制器连接，使用网络调试助手进行测试
- 2. 为udp server配置配方文件

系统IP 192.168.12.138

修改IP

端口 2000

状态 未连接

数量 0

端口 2001

状态 未连接

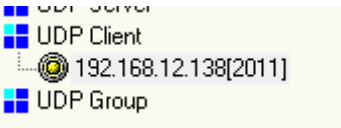
数量 0

端口 2011(UDP)

配方 system

更改配方

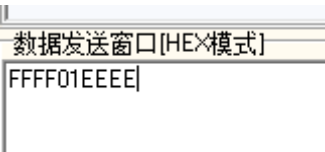
- 3. 使用调试助手创建udp client



- 4. 连接成功后将会收到周期性的机器人位姿数据

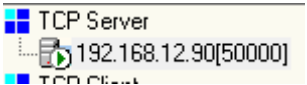
```
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
17:03:53收到数据: {FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE }
```

5. 向服务器发送FF FF 01 EE EE, 将会控制配置的对应的IO。



5.3.3.2 tcp/udp client

- 1. 使用网线将电脑和控制器连接，使用网络调试助手进行测试
- 2. 使用调试助手创建 tcp/udp 服务器



- 3. 通过 ui 创建 tcp/udp 客户端

添加连接

名称

tcp1

类型

TCP

IP地址

192.168.12.90

端口

50000

确定

I/O

寄存器

CCI

TCI

TCP/IP

Modbus/TCP

Profinet

服务器

系统IP 192.168.12.138 修改IP

端口 2000 状态 未连接 数量 0

端口 2001 状态 未连接 数量 0

端口 2011(UDP) 配方 system 更改配方

2000端口数据 清除

暂无数据

客户端

添加连接

tcp1

TCP/50000

已连接

N/A

断开

4. 为客户端选择配方文件

添加连接

tcp1	TCP/50000	已连接	system	断开
------	-----------	-----	--------	----

5. 调试助手端将会收到周期性的机器人位姿数据

数据接收及提示窗口

```
C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF
BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF
86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF
FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC
EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17
A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F
37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13
3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F
0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE
FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF 59 FC EA BE FE FE FF FF 39 2F 0B BF 86 87 13 3F BF BD 9D 3F 37 FC C8 BF 17 A0 1B BF
```

6. 向客户端发送FF FF 01 EE EE, 将会控制配置的对应的IO

中科新松有限公司 **SIASUN Co., Ltd**

地址：上海市浦东新区金藏路 257 号

电话：+086-021-50870608

传真：+086-021-63631363

邮编：201206

<http://www.siasun-in.com>

Add: NO.257 Jinzang Rd.

Pudong New District, Shanghai, China

Tel: +086-021-50870608

Fax: +086-021-63631363

Zip Code: 201206

<http://www.siasun-in.com>