

串行总线智能舵机

SCS15 用户手册

修订历史

日期	版本	更新内容
2016. 8. 19	V1. 01	1. 勘误修订 2. 增加运行速度控制参数



第一章 部件概述

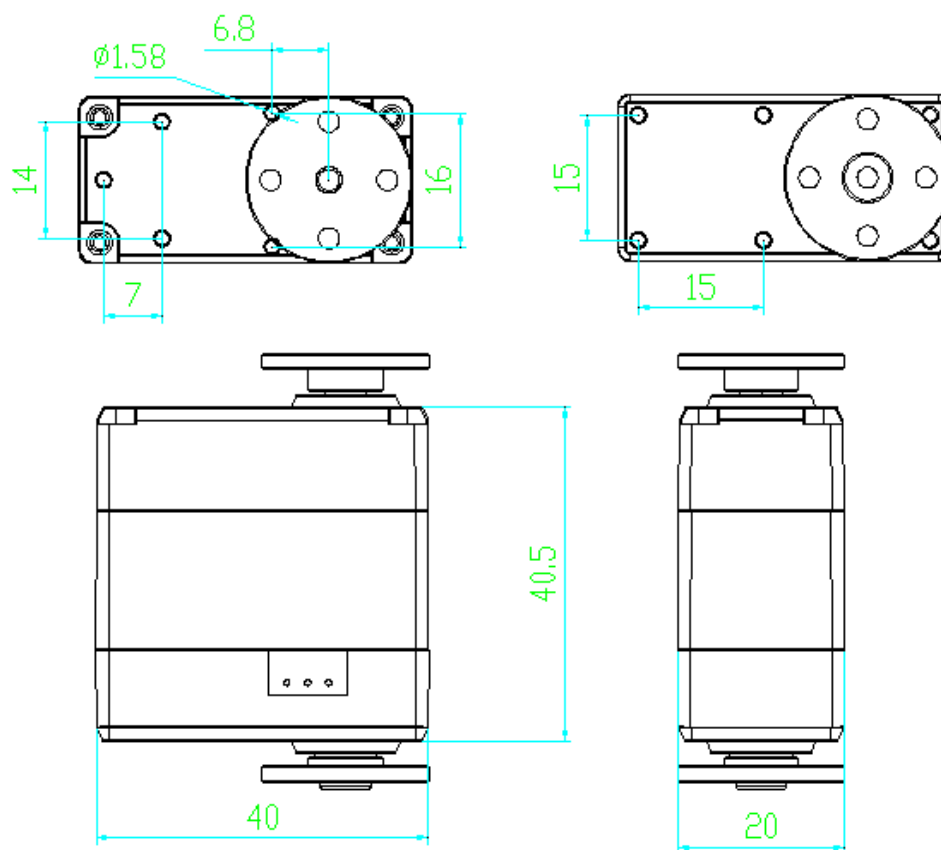
1.1 产品特性

SCS15 串行总线智能舵机集电机、伺服驱动、串行总线式通讯接口以及传感器为一体的集成伺服机构，主要用于微型机器人的关节、轮子、履带驱动，也可用于其他简单位置控制场合。

SCS15 的特点如下所示：

- ◆ 大扭矩：15Kg. cm
- ◆ 宽电压范围供电
DC 6V~9V
- ◆ 分辨率 0.19°
- ◆ 双轴安装方式，适合安装在机器人关节
- ◆ 高精度全金属齿轮组，双滚珠轴承
- ◆ 双连接端口易组菊花式布线
- ◆ 位置伺服控制模式下转动范围 0~200°
- ◆ 可设置为电机模式整周旋转，开环调速
- ◆ 总线连接理论可串联 253 个 ID 号
- ◆ 通讯波特率高达 1M
- ◆ 0.25KHz 的伺服更新率
- ◆ 采用串口总线 SCS 通讯协议
- ◆ 具备位置、温度、速度、电压反馈

1.2 结构尺寸



1.3 电气连接

1.3.1 引脚定义

SCS15 串行总线智能舵机电气接口如下图，两组引脚定义一致的接线端子可将舵机逐个串联起来。



1.3.2 舵机通讯方式

SCS15 采用异步串行总线通讯方式，理论最多可以连接 253 个机器人舵机。每个舵机可以设定不同的节点地址，多个舵机可以统一控制也可以单个独立控制。

SCS15 的通讯指令集开放，通过异步串行接口与用户的上位机(控制器或 PC 机)通讯，您可对其进行参数设置、功能控制。通过异步串行接口发送指令，SCS15 可以设置为电机控制模式或位置控制模式。在电机控制模式下，SCS15 可以作为直流减速电机使用，速度可调；在位置控制模式下，SCS15 拥有 0-200° 的转动范围，在此范围内具备高精度位置控制性能，转动速度可控制。

只要符合协议的半双工 UART 异步串行接口都可以和 SCS15 进行通讯，对 SCS15 进行各种控制。主要有以下两种形式：

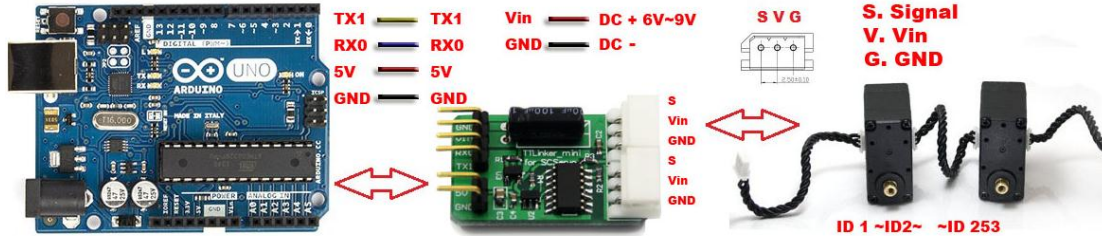
方式 1：通过 SCM-1 与 PC 机控制 SCS15

PC 机会将 SCM-1 识别为串口设备（驱动程序正常安装），上位机软件会通过串口发出符合协议格式的数据包，经 SCM-1 转发给 SCS15。SCS15 会执行数据包的指令，并且返回应答数据包。SCServoDebug.exe 是推荐使用的调试软件，您也可根据本手册提供的协议设计专用的 PC 端软件。



方式 2：通过专用控制器控制 SCS15

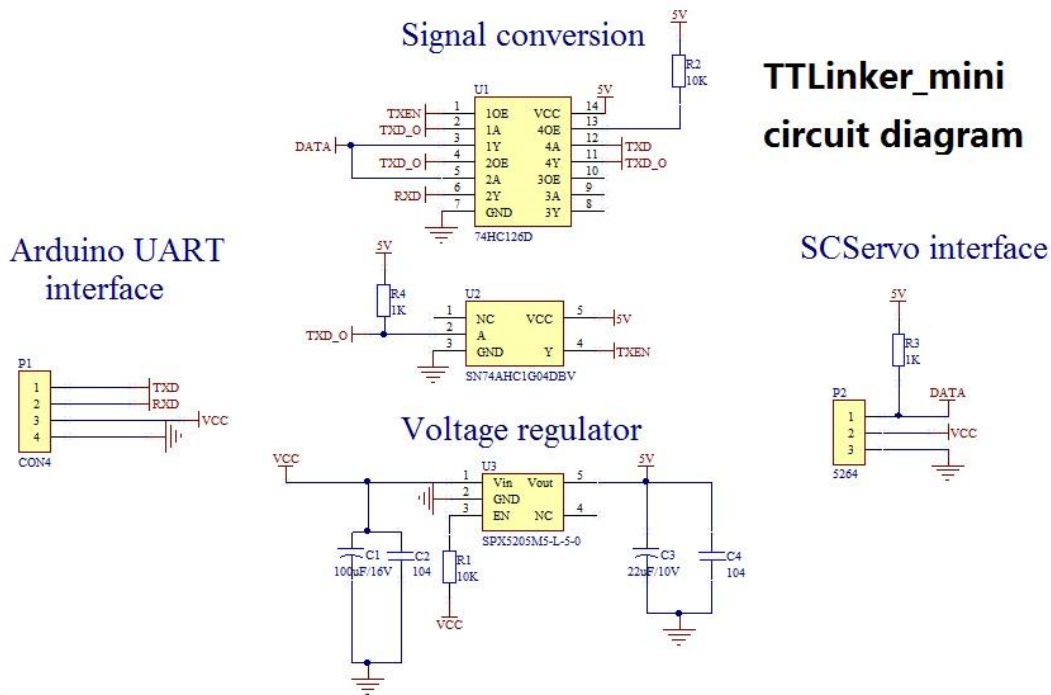
方式 1 可以快速简单地调试及控制 SCS 系列串行总线智能舵机、修改各种性能及功能参数。但是，这种方式离不开 PC 机，不能再加如传感器搭建独立自主控制的机器人。如可以设计专用的控制器或采用如 Arduino 通用控制器，通过控制器的 UART 端口控制舵机，1.3.3 小节给出了控制器 UART 接口信号转换原理图，可以依此设计出自己的控制器。



1.3.3 UART 接口原理图

SCS15 串行总线智能舵机实现半双工异步串行总线通讯，通讯速度可高达 1Mbps，且接口简单、协议精简。

在您自行设计的控制器中，用于和 SCS15 通讯的 UART 接口请参考如下图所示进行处理。



第二章 通讯协议

2.1 通信协议概要

控制器和舵机之间采用问答方式通信，控制器发出指令包，舵机返回应答包。

一个网络中允许有多个舵机，所以每个舵机都分配有一个 ID 号。控制器发出的控制指令中包含 ID 信息，只有匹配上 ID 号的舵机才能完整接收这条指令，并返回应答信息。

通信方式为串行异步方式，一帧数据分为 1 位起始位，8 位数据位和 1 位停止位，无奇偶校验位，共 10 位。

2.2 指令包

指令包格式：

字头	ID 号	数据长度	指令	参数	校验和
0XFF 0XFF	ID	Length	Instruction	Parameter1 ...Parameter N	Check Sum

字头：连续收到两个 0XFF，表示有数据包到达。

ID：每个舵机都有一个 ID 号。ID 号范围 0~253, 转换为十六进制 0X00~0XFD。

广播 ID：ID 号 254 为广播 ID, 若控制器发出的 ID 号为 254 (0XFE)，所有的舵机均接收指令, 除 PING 指令外其它指令均不返回应答信息 (多个舵机连接在总线上不能使用广播 PING 指令)。

数据长度：等于待发送的参数长度 N 加上 2，即 “N+2”。

参数：除指令外需要补充的控制信息。

校验和：校验和 Check Sum，计算方法如下：

$Check\ Sum = \sim (ID + Length + Instruction + Parameter1 + \dots Parameter\ N)$

若括号内的计算和超出 255, 则取最低的一个字节，“~”表示取反。

2.3 应答包

应答包是舵机对控制器的应答，应答包格式如下：

字头	ID	数据长度	当前状态	参数	校验和
0XFF 0XFF	ID	Length	ERROR	Parameter1 ...Parameter N	Check Sum

返回的应答包包含舵机的当前状态 ERROR，若舵机当前工作状态不正常，会通过这个字节反映出来，每一位的代表的信息如下：

BIT	名称	详细
BIT7	0	---
BIT6	0	---
BIT5	过载	位置模式运行时输出扭矩小于负载置1
BIT4	0	---
BIT3	0	---
BIT2	过热	温度超过指定范围置1
BIT1	0	---
BIT0	过压欠压	电压超过指定范围置1

若 ERROR 为 0，则舵机无报错信息。

若指令是读指令 READ DATA，则 Parameter1 ...Parameter N 是读取的信息。

2.4 指令类型

可用指令如下：

指令	功能	值	参数长度
PING (查询)	查询工作状态	0x01	0
READ DATA (读)	查询控制表里的字符	0x02	2
WRITE DATA (写)	往控制表里写入字符	0x03	不小于2
REGWRITE DATA(异步写)	类似于WRITE DATA, 但是控制字符写入后并不马上动作, 直到ACTION指令到达	0x04	不小于2
ACTION (执行异步写)	触发REG WRITE写入的动作	0x05	0
SYCNWRITE DATA (同步写)	用于同时控制多个舵机	0x83	不小于2
RESET (复位)	把控制表复位为出厂值	0x06	0

2.4.1 写指令 WRITE DATA

功能 写数据到总线舵机的控制表
长度 N+3 (N 为写入数据的长度)
指令 0X03
参数 1 数据写入段的首地址
参数 2 写入的第一个数据
参数 3 第二个数据
参数 N+1 第 N 个数据

例 1 把一个任意编号的 ID 设置为 1。

在控制表里保存 ID 号的地址为 3 (控制表见后文), 所以在地址 3 处写入 1 即可。发送指令包的 ID 使用广播 ID (0xFE)。

指令帧: 0XFF 0XFF 0XFE 0X04 0X03 0X03 0X01 0XF6

字头	ID	有效数据长度	指令	参数	校验和
0XFF 0XFF	0XFE	0X04	0X03	0X03 0X01	0XF6

因为采用广播 ID 发送指令, 所以不会有数据返回

2.4.2 读指令 READ DATA

功能 从舵机的控制表里读出数据
长度 0X04
指令 0X02
参数1 数据读出段的首地址
参数2 读取数据的长度

例 2 读取 ID 为 1 的舵机的当前位置。

在控制表里从地址 0X38 处读取二个字节。

指令帧: 0XFF 0XFF 0X01 0X04 0X02 0X38 0X02 0XBE

字头	ID	有效数据长度	指令	参数	校验和
0XFF 0XFF	0X01	0X04	0X02	0X38 0X02	0XBE

返回的数据帧: 0XFF 0XFF 0X01 0X03 0X00 0X20 0XDA

字头	ID	有效数据长度	工作状态	参数	校验和
0XFF 0XFF	0X01	0X04	0X00	0X00 0X20	0XDA

读出的数据是 0x0020，说明当前的位置为 32（0x0020）。

2.4.3 异步写指令 REG WRITE

REG WRITE 指令类似于 WRITE DATA，只是执行的时间不同。当收到 REG WRITE 指令帧时，把收到的数据储存在缓冲区备用，并把 Registered Instruction 寄存器（Address 0x40）置 1。当收到 ACTION 指令后，储存的指令最终被执行。

长度 N+3（N 为要写入数据的个数）

指令 0X04

参数 1 数据要写入区的首地址

参数 2 要写入的第一个数据

参数 3 要写入的第二个数据

参数 N+1 要写入的第 N 个数据

2.4.4 执行异步写指令 ACTION

功能 触发 REG WRITE 指令

长度 0X02

指令 0X05

参数 无

ACTION 指令在同时控制多个舵机时非常有用。

在控制多个舵机时，使用 ACTION 指令可以使第一个和最后一个舵机同时执行各自的动作，中间无延时。

对多个舵机发送 ACTION 指令时，要用到广播 ID（0xFE），因此，发送此指令不会有数据帧返回。

2.4.5 同步写指令 SYNC WRITE

功能 用于同时控制多个舵机。

ID 0xFE

长度 $(L + 1) * N + 4$ （L：发给每个舵机的数据长度，N：舵机的个数）

指令 0X83

参数 1 写入数据的首地址

参数 2 写入的数据的长度(L)

参数 3 第一个舵机的 ID 号

参数 4 写入第一个舵机的第一个数据

参数 5 写入第一个舵机的第二个数据

...

参数 L+3 写入第一个舵机的第 L 个数据

参数 L+4 第二个舵机的 ID 号

参数 L+5 写入第二个舵机的第一个数据

参数 L+6 写入第二个舵机的第二个数据

...

参数 2L+4 写入第二个舵机的第 L 个数据

...

不同于 REG WRITE+ACTION 指令的是实时性比它更高，一条 SYNC WRITE 指令可一次修改多个舵机的控制表内容，而 REG WRITE+ACTION 指令是分步做到的。尽管如此，使用 SYNC WRITE 指令时，写入的数据长度和保存数据的首地址必须相同即必须执行相同的动作。

例 对 4 个舵机写入如下的位置和速度

ID0: 位置: 0X010; 时间:1000 (0X3E8)

ID1: 位置: 0X220; 时间:1000 (0X3E8)

ID2: 位置: 0X030; 时间:1000 (0X3E8)

ID3: 位置: 0X220; 时间:1000 (0X3E8)

指令帧 : 0XFF 0XFF 0XFE 0X18 0X83 0X38 0X04 0X00 0X00 0X10 0X03 0XE8 0X01 0X02 0X20
0X03 0XE8 0X02 0X00 0X30 0X03 0XE8 0X03 0X02 0X20 0X03 0XE8 0X02

字头	ID	有效数据长度	指令	参数	校验和
0XFF 0XFF	0XFE	0X18	0X83	0x2A 0X04 0X00 0X00 0X10 0X03 0XE8 0X01 0X02 0X20 0X03 0XE8 0X02 0X00 0X30 0X03 0XE8 0X03 0X02 0X20 0X03 0XE8	0X02

因为采用广播 ID 发送指令，所以不会有数据返回。

2.4.6 复位指令 RESET

功能 把控制表里的数据复位为出厂值

长度 0X02

指令 0X06

参数 无

例 复位舵机，ID 号为 0。

指令帧: 0XFF 0XFF 0X00 0X02 0X06 0XF7

字头	ID	有效数据长度	指令	校验和
0XFF 0XFF	0X00	0X02	0X06	0XF7

返回的数据帧: 0XFF 0XFF 0X00 0X02 0X00 0XFD

字头	ID	有效数据长度	工作状态	校验和
0XFF 0XFF	0X00	0X02	0X00	0XFD

2.4.7 查询状态指令 PING

功能 读取舵机的工作状态

长度 0X02

指令 0X01

参数 无

例 3 读取 ID 号为 1 的舵机的工作状态

指令帧: 0XFF 0XFF 0X01 0X02 0X01 0XFB

字头	ID	有效数据长度	指令	校验和
0XFF 0XFF	0X01	0X02	0X01	0XFB

返回的数据帧: 0XFF 0XFF 0X01 0X02 0X00 0XFC

字头	ID	有效数据长度	工作状态	校验和
0XFF 0XFF	0X01	0X02	0X00	0XFC

第三章 内存控制表

3.1 内存控制表

机器人舵机本身的信息和控制参数形成了一张表，保存在其控制芯片的 RAM 和 EEPROM 区域。我们通过实时修改表里的内容，可以达到实时控制舵机的目的。这张表称为内存控制表，内容如下：

地址	命令项	读写	初始值	存储区域
0 (0x00)	--	--	--	EEPROM
1 (0x01)	--	--	--	
2 (0x02)	--	--	--	
3 (0x03)	软件版本 (H)	读	--	
4 (0x04)	软件版本 (L)	读	--	
5 (0x05)	ID	读/写	00 (0x00)	
6 (0x06)	波特率	读/写	00 (0x00)	
7 (0x07)	返回延迟时间	读/写	00 (0x00)	
8 (0x08)	应答状态级别	读/写	01 (0x01)	
9 (0x09)	最小角度限制 (H)	读/写	00 (0x00)	
10 (0x0A)	最小角度限制 (L)	读/写	00 (0x00)	
11 (0x0B)	最大角度限制 (H)	读/写	03 (0x03)	
12 (0x0C)	最大角度限制 (L)	读/写	255 (0xFF)	
13 (0x0D)	最高温度上限	读/写	80 (0x50)	
14 (0x0E)	最高输入电压	读/写	250 (0xFA)	
15 (0x0F)	最低输入电压	读/写	50 (0x32)	
16 (0x10)	最大扭矩 (H)	读/写	03 (0x03)	
17 (0x11)	最大扭矩 (L)	读/写	255 (0xFF)	
18 (0x12)	--	--	--	
19 (0x13)	卸载条件	读/写	00 (0x00)	
20 (0x14)	--	--	--	
21 (0x15)	P	读/写	15 (0x0F)	
22 (0x16)	--	--	--	
23 (0x17)	--	--	--	
24 (0x18)	最小PWM (H)	读/写	00 (0x00)	
25 (0x19)	最小PWM (L)	读/写	00 (0x00)	
26 (0x1A)	顺时针不灵敏区	读/写	02 (0x02)	
27 (0x1B)	逆时针不灵敏区	读/写	02 (0x02)	
28 (0x1C)	--	--	--	
29 (0x1D)	--	--	--	
30--39	--	--	--	
40 (0x28)	扭矩开关	读/写	00 (0x00)	RAM

41 (0x29)	--	--	--
42 (0x2A)	目标位置 (H)	读/写	--
43 (0x2B)	目标位置 (L)	读/写	--
44 (0x2C)	运行时间 (H)	读/写	00 (0x00)
45 (0x2D)	运行时间 (L)	读/写	00 (0x00)
46 (0x2E)	运行速度 (H)	读/写	00 (0x00)
47 (0x2F)	运行速度 (L)	读/写	00 (0x00)
48 (0x30)	锁标志	读/写	00 (0x00)
49--55	--	--	--
56 (0x38)	当前位置 (H)	读	?
57 (0x39)	当前位置 (L)	读	?
58 (0x3A)	当前速度 (H)	读	?
59 (0x3B)	当前速度 (L)	读	?
60 (0x3C)	当前负载 (H)	读	?
61 (0x3D)	当前负载 (L)	读	?
62 (0x3E)	当前电压	读	?
63 (0x3F)	当前温度	读	?
64 (0x40)	REG WRITE标志	读	00 (0x00)

若控制参数有“L”、“H”之分的命令，其范围为 0x00—0x3FF；参数只占一个字节的命令，其范围为 0x00—0xFE。

保存在 RAM 里的参数掉电后不会保存，保存在 EEPROM 里的参数，掉电后可以保存。“ -- ”表示不可修改参数。

详细描述如下：

0x05:

保存舵机的 ID 号。

0x06:

Address6 默认为 0，表示的波特率为 1m，可按下表把波特率修改为用户需要的其他波特率，其他的波特率会被恢复为 1M。

公式中 Address6 即为 0x06 地址保存的数据。波特率和相应的计算参数的对照如下表：

Address4	Hex	实际波特率	目标波特率	误差
0	0x00	1000000.0	1000000.0	0.000%
1	0x01	500000.0	500000.0	0.000%
2	0x02	250000.0	250000.0	0.000%
3	0x03	128000.0	128000.0	0.000%
4	0x04	115107.9	115200	0.079%
5	0x05	76923.0	76800	-0.160%
6	0x06	57553.9	57600	0.008%
7	0x07	38461.5	38400	-0.160%

0x07:

设置返回延迟时间，即当舵机收到一条需要应答的指令后，延迟多长时间应答可由您设置。时间范围：参数 (0~254) * 2US，若参数 250，即 500us 后应答；但参数为 0，表示以最短的时间应答。

0x08:

应答级别，设置舵机接收到数据后是否返回数据。

地址16	返回应答包
0	除读指令与PING指令外其它指令不返回应答包
1	对所有指令返回应答包

0x09~0x0C:

设置舵机可运行的角度范围，最小角度限制 = < 目标角度值 ≤ <= 最大角度限制值。



注意，最小角度限制值必须小于最大角度限制值。若目标角度值超过范围，则等于限制值。

0x0D

最高工作温度，如设置为 80 则最高温度为 80 度设置精度为 1 度

0x0E

最高工作电压，如高置为 85 则最高工作电压为 8.5V 设置精度为 0.1V

0x0F

最低工作电压，如高置为 45 则最低工作电压为 4.5V 设置精度为 0.1V

0x10~0x11

设置舵机的最大输出力矩。0X03FF 对应 SCS15 的最大输出能力。

0x13:

设置卸载条件。

BIT	功能
BIT7	--
BIT6	--
BIT5	如果设置为1，则发生过载时减少力输出
BIT4	--
BIT3	--
BIT2	如果设置为1，则发生过热时卸载
BIT1	--
BIT0	如果设置为1，则发生超过电压范围时卸载

以上若同时发生，遵行逻辑或的原则。

0x18~0x19:

PWM 占空比的最小输出值

0x1A~0x1B:

位置闭环的死区大小，顺时针与逆时针都设置为 1 则死区大小约为 0.38 度

0x28:

力矩输出开关，“1”开，“0”关。

0x2A~0x2B:

命令舵机运行至的位置，范围 0x0000—0x03FF, 0x0000 对应 0 度，0x03FF 对应 200 度，偏差±2%。

0x2C~0x2F:

设置舵机运行至目标位置的时间与速度(速度参数比时间参数优先，同时写入时间与速度参数，速度参数被选择为控制参数)，时间参数单位为(毫秒)，速度参数单位为(0.19 度/秒)如 1000 则速度为(1000*0.19)度/秒。设置为 0 时，对应与 SCS15 的最大速度 62RPM。

0x30:

锁功能位。若该位设置为 0 关闭锁保护，则对 EEPROM 区参数修改可以掉电保存。



注意，锁功能位设置为 0，SCS15 写速度会变慢，频繁对 EEPROM 区参数进行写入

操作会影响 SCS15 寿命。

0x40:

若有 REG WRITE 指令等待执行，则显示为 1，当 REG WRITE 指令执行完毕后显示为 0。

3.2 电机调速模式

SCS 系列机器人舵机可以切换为电机调速模式，可用于轮子、履带等周转的执行机构上。

把最小角度限制和最大角度限制(0x09~0x0C)都设置为 0，再给一个速度(0x2C~0x2D)，舵机就以电机调速模式转动起来。速度有大小和方向的控制方式，如下表所示：

BIT	11~15	10	9	8	7	6	5	4	3	2	1	0
VALUE	0	0/1	SPEED VALUE									

地址 0x2C~0x2D: BIT10 是方向位，为 0 逆时针转动，为 1 顺时针转动。BIT0~BIT9 为大小。