Serial Bus Smart Control servo

SCS15 Manual

Revision history

date	version	Update content
2016.8.19	V1.01	1. Corrigendum amendment 2. add speed control parameters
2016.12.21	V1.02	Delete protocol content

Part One Component overview

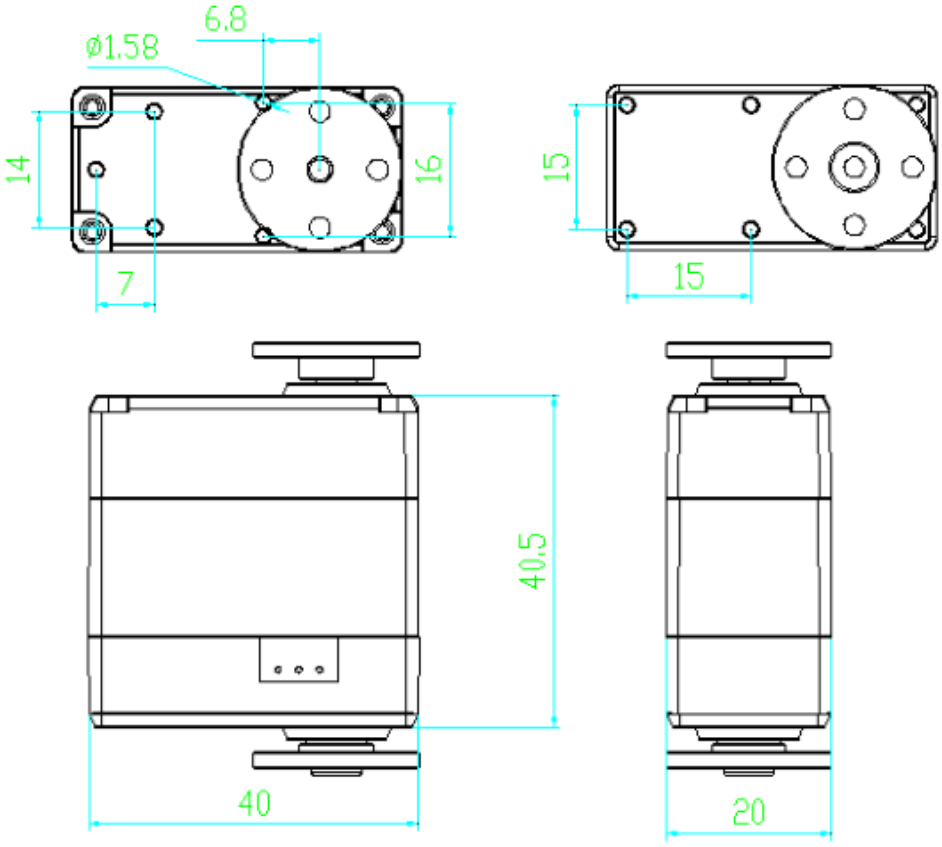
1.1 product feature

Serial bus smart control servo fully integrated Motor, servo drive, Serial bus communication interface and sensor together as one servo,Mainly used for micro robot joint, wheel, track drive,
Can also be used for other simple position control occasions.

The features of SCS15 are as follows

- High torque: 15Kg.cm
- Wide voltage range power supply DC 6V～9V
- Resolution:0.19°
- Double shaft install way, Fit for robot joint
- High precision all metal gear set, Two ball Bearing
- Dual connection port,daisy chain connection
- position servo mode can rotate 0-200°
- Can be set to motor mode, full cycle rotation,Open loop speed regulation
- The bus connection theory can be connected in series with 253 ID numbers
- The baud rate is high as 1M
- 0.25KHz Servo update rate
- Using serial bus SCS communication protocol
- With position, temperature, speed, voltage feedback

1.2 structure size



1.3 Electrical connection

1.3.1 Pin definition

Serial bus smart control The electrical interface is shown below, The two groups of pins define the same terminals and connect the servos one by one.

1.3.2 Servo communication mode

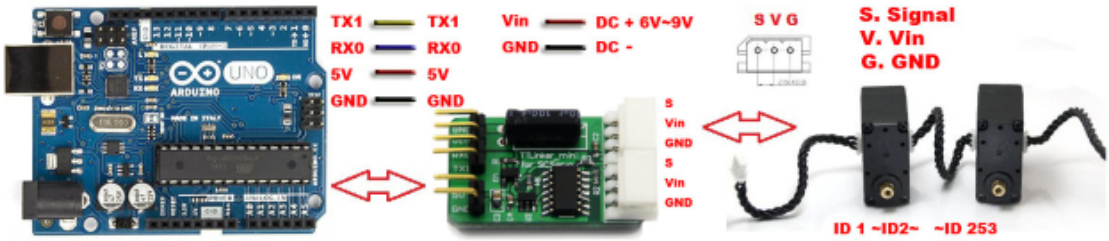
SCS15 Asynchronous serial bus communication mode, Theoretically, 253 robot servos can be connected at most。Each servo can set different node addresses, multiple servos can be unified control, or can be independently controlled.

The communication instruction set of SCS15 is open, communicating with the user's host computer (controller or PC machine) through the asynchronous serial interface. You can set the parameter and control the function. An instruction is sent via an asynchronous serial interface, and the SCS15 can be set to either a motor control mode or a position control mode. In the motor control mode, SCS15 can be used as DC motor, speed adjustable; in position control mode, SCS15 has the rotation range of 0-200 DEG, within this range with high precision position control performance, rotational speed control.
As long as the protocol conforms to the protocol, the half duplex UART asynchronous serial interface can communicate with the SCS15 and control the SCS15. There are two main forms:

Way 1: use SCM-1and Pccontrol SCServo

Computer will recognize SCM-1 as serial device (driver normal installation), the host computer software will be issued through the serial port in accordance with the protocol format of data packets, forwarded to SCS15 via SCM-1. The SCS15 will execute the packet instructions and return the response packets. SCServoDebug.exe is the recommended debugging software, and you can also design proprietary PC end software in accordance with the protocols provided by this manual.

Way 2: Use Dedicated controller to control SCServo

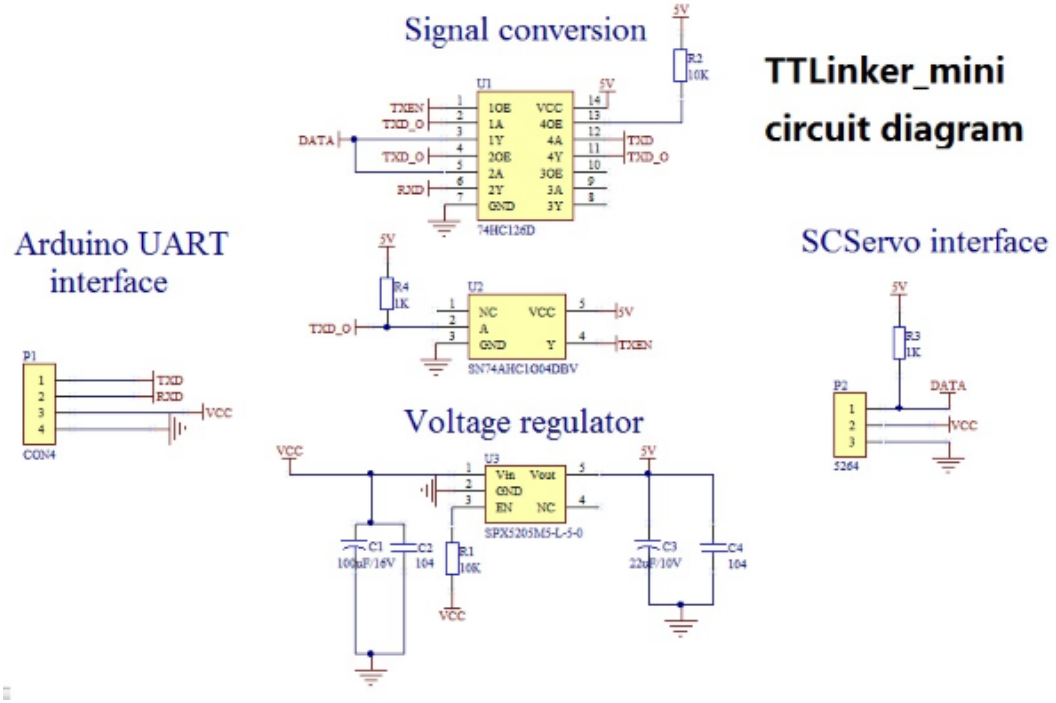


Mode 1 can quickly and easily debug and control the SCS serial bus intelligent servo system, and modify various performance and functional parameters. However, this approach can not be separated from the PC machine, and can not be added, such as sensors to build independent control of the robot. The design of special controllers such as can be used as Arduino or general controller, control steering controller through UART port, the 1.3.3 section gives the conversion principle of UART controller interface signal, can be designed their own controller.

1.3.3 UART Interface schematic diagram

SCS15 serial bus intelligent servo system realizes half duplex asynchronous serial bus communication, the communication speed can reach as high as 1Mbps, and the interface is simple and the protocol is simple.

In your self designed controller, the UART interface for communicating with the SCS15 is referred to as shown below。



Part 2 Memory control table

2.1 ERROR current state

The returned response packet contains the servo current status ERROR, If the servo current working state is abnormal, Will be reflected by this byte, Each Byte reprsent information is below:

BIT	Name	Detail
BIT7	0	---
BIT6	0	---
BIT5	Overload	In position mode, the output torque is less than the load when running
BIT4	0	---
BIT3	0	---
BIT2	Overheated	The temperature is over the specified range
BIT1	0	---
BIT0	Overvoltage and undervoltage	The voltage is over the specified range

If ERROR is 0, There is no wrong information about the servo。

2.2 Memory control table

The information and control parameters of the robot servo form a table which is stored in the RAM and EEPROM regions of the control chip. By modifying the contents constantly, we can control the servo constantly. This table is called the memory control table, which reads as follows:

Address	Command item	Read write	initial value	Storage area
0 (0x00)	--	--	--	EEPROM
1 (0x01)	--	--	--	
2 (0x02)	--	--	--	
3 (0x03)	software version (H)	read	--	
4 (0x02)	software version (L)	read	--	
5 (0x05)	ID	Read /write	00 (0x00)	
6 (0x06)	baud rate	Read /write	00 (0x00)	
7 (0x07)	Return delay time	Read /write	00 (0x00)	
8 (0x08)	Answer status level	Read /write	01 (0x01)	

9（0x09）	Minimum angle limit（H）	Read /write	00（0x00）	
10（0x0A）	Minimum angle limit（L）	Read /write	00（0x00）	
11（0x0B）	Maximum angle limit（H）	Read /write	03（0x03）	
12（0x0C）	Maximum angle limit（L）	Read /write	255（0xFF）	
13（0x0D）	Maximum temperature limit	Read /write	80（0x50）	
14（0x0E）	Highestinput voltage	Read /write	250（0xFA）	
15（0x0F）	Lowest input voltage	Read /write	50（0x32）	
16（0x10）	Max torque（H）	Read /write	03（0x03）	
17（0x11）	Max torque（L）	Read /write	255（0xFF）	
18（0x12）	High voltage flag bit	Read /write	00（0x00）	
19（0x13）	Unload condition	Read /write	37（0x25）	
20（0x14）	LED alarm condition	Read /write	37（0x25）	
21（0x15）	P	Read /write	15（0x0F）	
22（0x16）	D	Read /write	00（0x00）	
23（0x17）	I	Read /write	00（0x00）	
24（0x18）	Minimum PWM（H）	Read /write	00（0x00）	
25（0x19）	Minimum PWM（L）	Read /write	00（0x00）	
26（0x1A）	Clockwise insensitive range	Read /write	02（0x02）	
27（0x1B）	Counterclockwise insensitive region	Read /write	02（0x02）	
28（0x1C）	Integral restriction（L）	Read /write	00（0x00）	
29（0x1D）	Integral restriction（H）	Read /write	00（0x00）	
30（0x1E）	Differential sampling coefficient	Read /write	00（0x00）	
31—39	--	--	--	
40（0x28）	Torque switch	Read /write	00（0x00）	RAM
41（0x29）	--	--	--	
42（0x2A）	Target position（H）	Read /write	--	
43（0x2B）	Target position（L）	Read /write	--	

44 (0x2C)	Running time (H)	Read /write	00 (0x00)	
45 (0x2D)	Running time (L)	Read /write	00 (0x00)	
46 (0x2E)	Running speed (H)	Read /write	00 (0x00)	
47 (0x2F)	Running speed (L)	Read /write	00 (0x00)	
48 (0x30)	Lock sign	Read /write	00 (0x00)	
49—55	--	--	--	
56 (0x38)	Current position (H)	Read	?	
57 (0x39)	Current position (L)	Read	?	
58 (0x3A)	Current speed (H)	Read	?	
59 (0x3B)	Current speed (L)	Read	?	
60 (0x3C)	Current load (H)	Read	?	
61 (0x3D)	Current load (L)	Read	?	
62 (0x3E)	Current voltage	Read	?	
63 (0x3F)	Current temperature	Read	?	
64 (0x40)	REG WRITE sign	Read	00 (0x00)	

If the control parameter has the command of "L" and "H", its range is 0x00 - 0x3FF; the parameter takes only one byte of command, and its range is 0x00 - 0xFE.

The parameters stored in RAM will not be saved after power down. The parameters stored in the EEPROM can be saved after power down. "-" indicates that parameters cannot be modified.

Details are described below:

0x06:

Address 0x06 defaults to 0, indicating the baud rate is 1M, you can press the table to change the baud rate to the user's other baud rates, and the other baud rate will be restored to 1M. The baud rate and the corresponding computed parameters are listed below:

Address6	Hex	Actual baud rate	Target baud rate	error
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:

Set the back delay time, that is, how long the delay is when the servo receives a command that needs to be answered, and the response is set by you. Time range: parameter (0~254) *2US, if the parameter 250, that is, after 500us response, but the parameter is 0, which means the shortest response time.

0x08:

The response level sets whether the data is returned after the servo has received the data。

Address 16	Return response packet
0	Other instructions do not return the acknowledge packet except the read instruction and the PING instruction
1	Returns the response packet to all instructions

0x09～0x0C:

Sets the angle range that the actuator can run, Minimum angle limit≤Target angle value≤Maximum angle limit

Note that the minimum angle limit must be less than the maximum angle limit. If the target angle exceeds the range, it is equal to the limit value.

0x0D

The maximum operating temperature, such as setting to 80, the highest temperature is 80 degrees, and the setting accuracy is 1 degrees

0x0E

The highest operating voltage, such as the high set to 85, the maximum working voltage of 8.5V, the set accuracy of 0.1V

0x0F

The minimum operating voltage, such as the high set to 45, the minimum operating voltage of 4.5V, the set accuracy of 0.1V

0x10～0x11

Set the maximum output torque of the stservo. 0X03FF corresponds to the maximum output torque of the SCS15.0x13:

Set uninstall condition, /LED alarm condition。

BIT	function
BIT7	--
BIT6	--
BIT5	If set to 1, the overload output /LED alarm occurs when overload occurs
BIT4	--
BIT3	--
BIT2	If set to 1, release the torque /LED alarm when overheating occurs
BIT1	--
BIT0	If set to 1, the torque /LED is released when the voltage range exceeds the voltage range

To act in accordance with logic or principles. LED alarm condition (0X14) set to 0, turn off LED, otherwise open LED。

0x1A～0x1B:

The size of dead zone position loop, clockwise and counterclockwise are set to 1 big dead zone is about 0. less 38

0x28:

Torque output : “1” turn on, “0” turn off。

0x2A～0x2B:

Target position, range 0x0000 - 0x03FF, 0x0000 corresponds to 0 degrees, 0x03FF corresponds to 200 degrees, deviation + 2%。

0x2C～0x31:

Set time and speed servo operation to the target location (velocity parameters than the time parameters of priority, and the writing time and speed parameters, velocity parameters are selected as control parameters), time parameter unit (MS), velocity parameters unit (0.19 / sec) as the 1000 speed for (1000*0.19) deg / sec. Set to 0, corresponding to the maximum speed SCS15 of 62RPM。

0x30：

Lock function bit. If this bit is set to 0, turn off the lock protection, then the EEPROM zone parameter changes can be saved out of power。

Please note, lock function byte set to 0, SCS15 write speed turn slow, Frequent write operations on the EEPROM parameter affect the life of the SCS15.

0x40：

If the REG WRITE command waits for execution, it is shown as 1, and when the REG WRITE instruction is executed, it is shown as 0。

2.3 Motor speed regulation mode

SCS series robot servo can be switched to motor speed regulation mode. It can be used in rotating mechanism such as wheel and crawler.

Limit the minimum angle and maximum angle（0x09～0x0C）to 0, give a speed（0x2C～0x2D）, The servo will rotate in the motor speed regulation mode。The speed is controlled by size and direction, as shown in the following table:

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

Address 0x2C～0x2D：BIT10 It’s the direction bit, if 0 it is turn anticlockwise , if 1it is Clockwise rotation。BIT0～BIT9 is value。