FEETECH SCS15 Bus Smart Control Servo

Product Specifications

- Product Name: Feetech Serial Bus Smart Control Servo
- Communication Protocol: Serial Bus Intelligent
- Series Compatibility: SCS and SMS series of servos
- SCS Series:

    - Communication: TTL level, single bus
    - Physical Connection: Three lines (two positive and negative poles of power supply)
- SMS Series:
    - Main Control Core: ARM 32-bit single-chip computer
    - Position Induction: 360-degree 12-bit precision magnet induction angle scheme
    - Communication Level: RS-485 mode

Product Usage Instructions

Communication Protocol Overview:
The Feetech Serial Bus Smart Control Servo communication protocol is designed for SCS and SMS series servos. Each servo in the network is assigned a unique ID number for communication.

Instruction Packet Format:
The instruction packet consists of the following components:

- Initial: 0XFF 0XFF indicating the start of a data packet
- ID No.: Unique ID number for each servo (0 to 253)
- Broadcast ID: ID No. 254 for broadcast instructions
- Data Length: Parameter count + 2
- Instruction: Operating Function Code
- Parameters: Additional control information, supporting up to two-byte parameters

Communication Mode:
The communication mode is serial asynchronous with a frame structure of start bit, data bits, and stop bit. Parity bits are not used, totaling 10 bits.

Memory Table Usage:
For parameters represented by two bytes, the SCS series and SMS series have different byte order conventions. Refer to the specific model's memory table for actual control functions.

Serial Bus Smart Control Servo Communication Protocol Manual

Revision history

Date	Version	Update content	
2017.03.01	V1.00	Initial formulation	Alex lee
2019.02.19	V1.01	Modified Description, Universal SCS and SMS Series servo	Alex lee

Summary of Communication Protocol

- The communication protocol of the FEETECH Serial Bus Intelligent servo is mainly applicable to FEETECH SCS and SMS series of servos. SCS series servo adopts TTL level and single bus (a signal line time-sharing multiplexing transmission and receiving data signal) communication connection, physical connection is three lines, including two positive and negative poles of power supply. The
- SMS series servo adopts an ARM 32-bit single-chip computer as the main control core, and position induction adopts a 360-degree 112-bit precision magnet induction angle scheme. The communication adopts RS-485 mode with strong anti-jamming ability. The communication still adopts asynchronous duplex, and the sending and receiving signals are processed asynchronously.
- Question-and-answer communication is adopted between the controller and the servo. The controller sends out the instruction package, and the servo returns to the response package.
- Multiple servos are allowed in a bus control network, so each servo is assigned a

unique ID number in the network. The control command issued by the controller contains ID information. Only the servo matching the ID number can receive the command completely and return the response information.

- The communication mode is serial asynchronous. A frame of data is divided into 1 1-bit start bit, 8 8-bit data bits, and 1-bit stop bit. There are no parity bits; a total of 10 bits.
- The difference between the SCS series and the SMS series communication protocols is that two bytes represent the high byte and the low byte, respectively, when some parameters of the memory table are in the range of two bytes. Among them, the parameters of the SCS series are in the address of the memory table after the high byte and the low byte after the high byte, while the SMS series is in the low byte after the high byte. In addition, each servo has slightly different functions, so the actual control should refer to the memory table of the specific model.

Instruction packet

Instruction package format

initial	ID No.	Data Length	Instruction / Command	Parameter	Checksum
0XFF 0XFF	ID	Length	Instruction	Parameter1 …Parameter N	Check Sum

Initial Continuous receipt of two 0XFFs indicating the arrival of data packets

- ID No. Each servo has an ID number. ID number ranges from 0 to 253, converted to hexadecimal 0X00~0XFD
- Broadcast ID: ID No. 254 is a broadcast ID. If the ID number issued by the controller is 254 (0XFE), all the Servos receive instructions, and no response information is returned except PING instructions (multiple servos can not use roadcast PING instructions on the bus).
- Data length: equal to the parameter N to be sent plus 2, that is, “N + 2″

- Instruction: Packet Operating Function Code, see Instruction Type 1.3

PARAMETERS:

- In addition to the additional control information required by the instructions, the parameters support a maximum of two-byte parameters to represent a memory value. The byte order refers to the manual memory control table for servo usage (different types of servos have different byte orders).
- Check sum: Check sum and Check Sum, the calculation method is as follows: Check Sum = ~ (ID + Length + Instruction + Parameter1 + … Parameter N). If the sum in parentheses exceeds 255, the lowest byte will be taken, and "~" means reverse.

Reply Packet

Reply packet is the servo's reply to the controller. Reply packet format is below:

initial	ID No	Data Length	current state	Parameter	Check sum
0XFF 0XFF	ID	Length	ERROR	Parameter1 …Parameter N	Check Sum

- The returned response package contains the current status ERROR of the servo.
- If the current status of the servo is not normal, it will be reflected through this byte (the meaning of each status is detailed in the manual memory control table). If ERROR is 0, the servo will have no error information.
- If the instruction is a read instruction (READ DATA), then Parameter 1… Parameter N is the read information.

Instruction type

The following instructions are available for Feetech Serial Bus Intelligent servo Communication Protocol

instruction	function	value	Parameter length
PING	Query the working status	0x01	0
READ DATA	Query the Characters in the Control Table	0x02	2
WRITE DATA	Write characters into the control table	0x03	≥1
REGWRITE DATA	Similar to WRITE DATA, the control character does not act immediately after writing until the ACTION instruction arrives.	0x04	Not less than 2
ACTION	Actions that trigger REG WRITE writes	0x05	0
SYCNWRITE DATA	For simultaneous control of multiple servos	0x83	Not less than 2
RESET	Reset the control table to the factory value	0x06	0

1Query status instruction PING

- Function: Read the working state of the servo
- Length 0X02
- Instruction 0X01
- Parameter no
- The PING command uses the broadcast address, and the steering gear also returns the response information.

Example 1 reads the working state of the steering gear with ID number 1

- **Instruction frame:** FF FF 01 02 01 FB `(sent in hexadecimal)

initial	ID	Effective data	instructi	Check

			length	on	Sum	
	0XFF 0XFF	0X01	0X02	0X01	0XFB	

Data frame returned: FF FF 01 02 00 FC (hexadecimal display)

initial	ID	Effective data length	working condition	Check Sum
0XFF 0XFF	0X01	0X02	0X00	0XFC

READ DATA

Function reads data from the servo memory control table

- Length * 0X04
- Instruction 0X02
- Parameter 1. Head address of the read-out segment of data
- Parameter 2. Length of read data
- Example 2: Read the current position of the servo with ID 1 (low byte before, high byte after).
- Two bytes are read from address 0X38 in the control table. Instruction frame: FF FF 01 04 02 38 02 BE (sent in hexadecimal)

initial	ID	Effective data length	instruction	Parameter	Check Sum

0XFF 0XFF	0X01	0X04	0X02	0X38 0X02	0XBE

Data frame returned: FF FF 01 04 00 18 05 DD (hexadecimal display)

initial	ID	Effective data length	working condition	Parameter	Check Sum
0XFF 0XFF	0X01	0X04	0X00	0X18 0X05	0XDD

Read out two byte data: low byte L 0X18 high byte H 0X05

- Two-byte synthesis of 16-bit data 0X0518, using decimal representation of the current location of 1304.

WRITE DATA

- Function. Write data to the servo memory control table
- Length N+3 (N is the parameter length)
- Instruction 0X03
- Parameter 1. Head address ofthe the data write segment
- Parameter 2: The first data written
- Parameter 3. Second data
- Parameter N+1 Number N Data
- Example 3 sets an ID of any number to 1.
- The address ofthe ID number is 5 in the control table, so write 1 at address 5. The ID of the sending instruction package uses the broadcast ID (0xFE).
- Instruction frame: FF FF FE 04 03 05 01 F4 (sent in hexadecimal)

initial	ID	Effective data length	instruc tion	Parameter	Check Sum

0XFF 0XFF	0XFE	0X04	0X03	0X05 0X01	0XF4

Because broadcasting ID is used to send instructions, there will be no data return. In addition, the memory table EPROM has a protective lock switch, which needs to be turned off before modifying the ID; otherwise, the sample ID number will not be saved when the power is off. For detailed operation, please refer to the memory table or operation manual of the specific steering gear type.

Example 4 controls the ID1 servo to rotate to 2048 at a speed of 1000 seconds.

In the control table, the first address of the target location is 0X2A, so six consecutive bytes of data are written at the address 0X2A, namely position data 0X0800 (2048), time data 0X0000 (0), speed data 0X03E8 (1000). The ID of the sending instruction package uses a non-broadcast ID (0xFE), so the servo will return to the status package when the instruction is received.

Instruction frame: FF FF 01 09 03 2A 00 08 00 E8 03 D5 (sent in hexadecimal)

Instruction frame: FF FF 01 09 03 2A 00 08 00 E8 03 D5 (sent in hexadecimal)

initial	ID	Effective data length	instruction	Parameter	Check Sum
0XFF 0XFF	0X01	0X09	0X03	0X2A	0XD5
				0X00 0X08	
				0X00 0X00	
				0XE8 0X03	

Data frame returned: FF FF 01 02 00 FC (hexadecimal display

initial	ID	Effective data length	working condition	Check Sum

- 0XFF 0XFF 0X01 0X02 0X00 0XFC

The return working state is 0, indicating that the servo has received the instructions correctly and correctly and has begun to execute them.

REG WRITE

The REG WRITE instruction is similar to the WRITE DATA except that the execution time is different. When the REG WRITE instruction frame is received, the received data is stored in the buffer reserve, and the Registered Instruction Register is set to 1. When the ACTION instruction is received, the stored instruction is finally executed.

- Length N+3 (N is the number of data to be written)
- Instruction 0X04
- Parameter 1. The header address of the area where the data is to be written, Parameter 2. The first data to be written
- Parameter 3. The second data to be written
- Parameter N+:1 The Nth Data to Write
- Example: 5 Control ID1 to ID10 servo to rotate to 2048 position at 1000 per second. Onthe only ID in the following instruction package receives instructions on the bus and returns. Other ID numbers are not returned on the bus.
- ID 1 Asynchronous Write Instruction Pack FF FF 01 09 04 2A 00 08 00 00 E8 03 D4 ID 1 Return Pack FF FF 01 02 00 FC
- ID 2 Asynchronous Write Instruction Pack   FF FF 02 09 04 2A 00 08 00 00 E8 03 D3 ID 3 Asynchronous Write Instruction Pack FF FF 03 09 04 2A 00 08 00 00 E8 03 D2 ID 4 Asynchronous Write Instruction Pack FF FF 04 09
- 04 2A 00 08 00 00 E8 03 D1 ID 5 Asynchronous Write Instruction Pack FF FF 05 09 04 2A 00 08 00 00 E8 03 D0 ID 6 Asynchronous Write Instruction Pack FF FF 06 09 04 2A 00 08 00 00 E8 03 CF ID 7 Asynchronous Write
- Instruction Pack FF FF 07 09 04 2A 00 08 00 00 E8 03 CE ID 8 Asynchronous Write Instruction Pack FF FF 08 09 04 2A 00 08 00 00 E8 03 CD ID 9 Asynchronous Write

Instruction Pack FF FF 09 09 04 2A 00 08 00 00 E8 03

- CC ID10 Asynchronous Write Instruction Pack FF FF 0A 09 04 2A 00 08 00 00 E8 03 CB

Executing Asynchronous Write Instruction ACTION

Function trigger REG WRITE instruction

- Length 0X02
- Instruction 0X05
- Parameter no

ACTION instruction is very useful for controlling multiple servos at the same time.

- When controlling multiple servos, the ACTION command enables the first and last servos to perform their respective actions simultaneously without delay.
- When the action command is sent to multiple servos, the broadcast ID (0xFE) is used, so no data frame will be returned when the command is sent.
- Example 6: After issuing the asynchronous writing instructions that control ID1 to ID10servoso t,o rotate the 2048 position at a speed of 1000 seconds, the following instruction packages (FF FF FE 02 05 FA) need to be sent when the asynchronous writing instructions need to be executed. All servos on the bus receive this instruction and run the asynchronous writing instruction received before.

SYNC WRITE

- Function used to control multiple servos
- ID 0XFE
- Length (L + 1) * N + 4 (L: Length of data sent to each servo, N: Servo Number)
- Instruction 0X83
- Parameter 1: Head address of write data
- Parameter 2 Length of write data(L)
- Parameter 3 First servo Number
- Parameter 4 Write the first data of the first servo
- Parameter 5 Write the L data of the first servo

- Parameter L+3 Write the second data of the first servo Parameter L+4
- The second Servo ID number
- Parameter L+5 Write the first data of the second servo
- Parameter L+6 Write the second data of the second servo
- Parameter 2L+4 Write the L data of the second servo
- Unlike the REG WRITE + ACTION instruction, the real-time performance is higher. A SYNC WRITE instruction can modify the control table contents of multiple servos at one time, while the REG WRITE + ACTION instruction can be implemented step by step.
- Nevertheless, when using SYNC WRITE instructions, the length of the data written must be the same as the first address of the data saved.
- Example 77 Writing position 0X0800, time 0X000,0 and speed 0X03E8 for ID1-ID4 with four servo header addresses 0X2A (low byte in front, high node in back)

Instruction free: FF FF FE 20 83 2A 06 01 00 08 00 00 E8 03 02 00 08 00 00 E8 03 03 00 08 00 00 E8 03 04 00 08 00 00 E8 03 58 (Send in hexadecimal)

initial	ID	Effective data length	instructions	**Parameter**	Check Sum
0XFF 0XFF	0XFE	0X20	0X83	OX2A OX06 OX01 OX00 OX08 OX00 OX00 OXE8 OX03 OX02 OX00 OX08 OX00 OX00 OXE8 OX03 OX03 OX00 OX08 OX00 OX00 OXE8 OX03 OX04 OX00 OX08 OX00 OX00 OXE8 OX03	0X58

Because broadcasting ID is used to send instructions, no data is returned.

RESET Instruction

- Function:n Reset the specific data in the memory control table (specific Servo type is used)
- length 0X02
- Instruction 0X06
- Parameter NO
- For example, le rest et servo ID number is 0
- Instruction frame FF 01 02 06 F6 (Send in hexadecimal)

initial	ID	Effective data length	instructions	Check Sum
0XFF 0XFF	0X00	0X02	0X06	0XF7

Returned data frame FF FF 01 02 00 FC Send in hexadecimal

initial	ID	Effective data length	working condition	Check Sum
0XFF 0XFF	0X01	0X02	0X00	0XFC

- **TEL** +86-755-8933-5266
- **FAX** +86-755-2696-6318
- www.feetechrc.com
- www.feetech.cn

FAQs

Q: Can multiple servos be controlled in a bus network?

A: Yes, multiple servos can be controlled in a bus network. Each servo is assigned a unique ID number for communication.

Q: What is the difference between SCS and SMS series communication protocols?

A: The main differences lie in the physical connection, communication level, and memory table parameter conventions. SCS uses TTL level with single bus communication, while SMS employs RS485 mode with 32-bit ARM control core.

Documents / Resources

 深圳飞特模型有限公司 Serial Bus Smart Control servo Communication Protocol Manual Revision history Date / Version / Update content 2017.03.01 / V1.00 / Initial formulation / Alex lee 2019.02.19 / V1.01 / Modified Description, Universal SCS and SMS Series servo / Alex lee TEL: +86-755-8933-5266 FAX: +86-755-2696-6318 www.feetechrc.com www.feetech.cn Page 1 of 9	FEETECH SCS15 Bus Smart Control Servo [pdf] Instruction Manual SCS15 Bus Smart Control Servo, SCS15, Bus Smart Control Servo, Smart Control Servo, Control Servo

References

- User Manual

FEETECH

Bus Smart Control Servo, Control Servo, FEETECH, SCS15, SCS15 Bus Smart Control Servo, Smart Control Servo

Leave a comment

Your email address will not be published. Required fields are marked *

Comment *

Name

Email

Website

□ Save my name, email, and website in this browser for the next time I comment.

Post Comment

Search:

e.g. whirlpool wrf535swhz

Search

Manuals+ | Upload | Deep Search | Privacy Policy | @manuals.plus | YouTube

This website is an independent publication and is neither affiliated with nor endorsed by any of the trademark owners. The "Bluetooth®" word mark and logos are registered trademarks owned by Bluetooth SIG, Inc. The "Wi-Fi®" word mark and logos are registered trademarks owned by the Wi-Fi Alliance. Any use of these marks on this website does not imply any affiliation with or endorsement.