NT86 MODBUSRTU USER MANUAL

Shenzhen Rtelligent Mechanical Electrical Technology Co., Ltd.

Chapter 1 Drive Description

1.1 Product Introduction

The NT86 is a high-performance bus-controlled stepper motor driver that integrates an intelligent motion controller function with built-in S-shaped acceleration/deceleration commands to independently set acceleration and deceleration. Real-time control of the drive and motor by running the Modbus protocol over the RS485 network.

1.1.1 Features

- Programmable small size stepper motor driver
- Working voltage DC: AC 18-80V / DC24- 100V, power greater than 150W
- Control mode: Modbus/RTU
- Communication method: RS485
- Maximum phase current output: 7.2A/phase (sinusoidal peak)
- Digital IO port:

6-channel optically isolated digital signal input: IN1, IN2 are 5V differential inputs, can also be connected to 5V single-ended input; IN3 ~ IN6 is 24V single-ended input, common anode connection;

2 channels of optically isolated digital signal output, maximum withstand voltage of 30V, maximum sinking or pulling current of 100mA, common cathode connection.

1.2 Safety instructions

The transportation, installation, use or maintenance of this product must be carried out by personnel who are professionally qualified and familiar with the above operations.

In order to minimize potential safety hazards, you should comply with all local and national safety regulations when using this equipment. Different areas have different safety regulations. You should ensure that the equipment is installed and used in accordance with your region. specification.

System errors can also result in equipment damage or personal injury. We do not guarantee that this product is suitable for your specific application and we cannot be held responsible for the reliability of your system design.

Be sure to read all relevant documents before installation and use. Improper use may result in equipment damage or personal injury. Please strictly observe the relevant technical requirements during installation. Be sure to confirm the grounding of each device in the system. Ungrounded systems cannot guarantee safe use of electricity.

Some components inside the product may be damaged by external static electricity. Before touching the product, the operator should ensure that there is no static electricity and avoid contact with objects that are easily electrostatically charged (chemical fiber, plastic film, etc.).

If your device is placed in the control cabinet, please close the control cabinet cover or the door during operation, otherwise it may cause equipment damage or personal injury.

It is strictly forbidden to hot plug the cable while the system is running. The arc generated by hot plugging may cause harm to operators and equipment.

Wait at least 3 seconds after turning off the power to touch the product or remove the wiring. Capacitive devices may still store dangerous electrical energy after power failure and require some time to release. To ensure safety, you can measure it with a multimeter before touching the product.

Please observe the important safety instructions presented in this manual, including clear warning symbols for potential safety hazards, and read and become familiar with these instructions before installation, operation and maintenance. The purpose of this paragraph is to inform users of the necessary safety instructions and to reduce the risk of personal injury and equipment safety. Miscalculations of the importance of safety prevention can cause serious damage or render the device unusable.

1.3 Preparation before starting

Before you start, please confirm that you have the following components:

- A stepper motor that matches the drive
- A small slotted screwdriver for tightening the connector screws
- A computer with Microsoft Windows XP/Vista/Windows 7/Windows 8/Windows 10 (32-bit or 64-bit) operating system installed
- NTConfigurator software (can be downloaded from the official website)
- Tip: When the first drive is connected to the RS-485 communication port of the computer or controller, the network cable can be cut into two segments. One section is used to connect the drive to the RS-485 communication port of the computer or controller, and the other section can be used to connect the terminal matching resistor to the RS-485 communication port at the end of the last driver on the bus.

1.3.1 Installing NTConfigurator

- Download and install the NTConfigurator software;
- Click Start / All Programs / RETELLIGENT / NTConfigurator to run the software;
- Connect the drive to your computer using a communication cable.

1.3.2 Connecting the power supply

- Connect the driver to the DC power supply: V+ is connected to the DC power supply positive terminal, and V-connected DC power supply is negative.
- Secure the connection between the drive base and the ground through the grounding screw

1.3.3 Connecting the motor

If the motor you are using is a Rtelligent brand stepper motor, connect the four wires of black, green, red and blue to the A+, A-, B+, B- ports of the drive.

The default drive motor model is a two-phase stepper motor. If the user needs to match the three-phase stepper motor, first modify the motor type through the commissioning software and then connect the three-phase stepper motor.

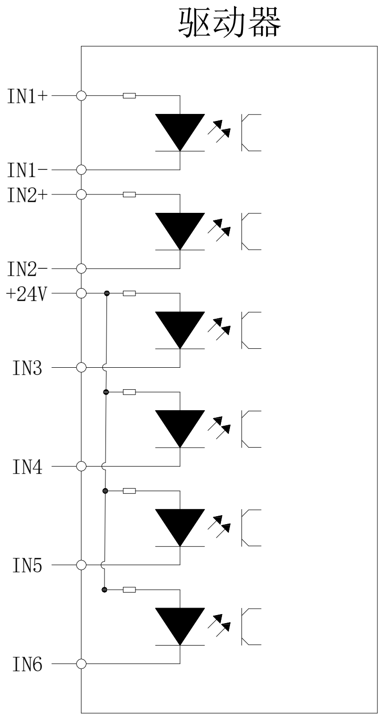
1.4 Digital input and output port

1.4.1 Digital input and output ports

The NT86 stepper driver has 6 digital inputs and 2 digital outputs. The digital input and output ports can be freely configured with various functions according to their own application requirements.

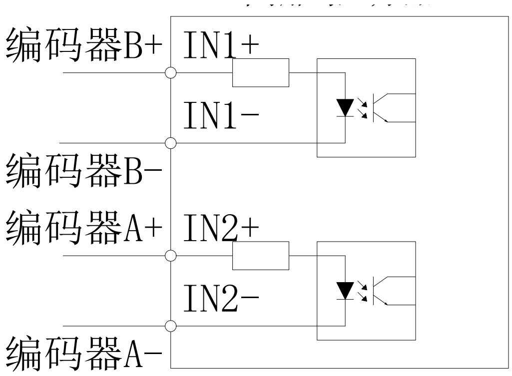
- **Note: IN1+/IN1-, IN2+/IN2- are 5V input terminals. Do not directly connect the input signal higher than this voltage, otherwise the drive will be damaged!**

The schematic diagram of the input port is shown below, and the user can perform system wiring according to the schematic diagram.



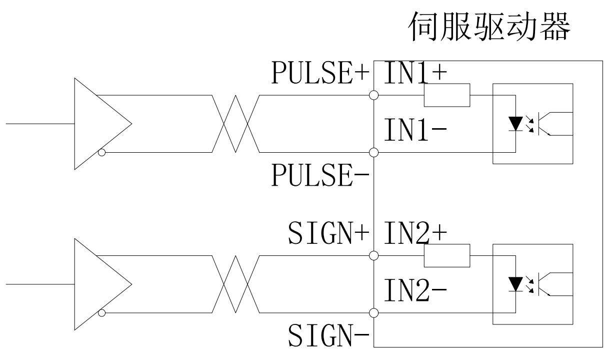
a) IN1+/IN1-, IN2+/IN2- differential input terminals

1. The external motor encoder forms a closed loop system:

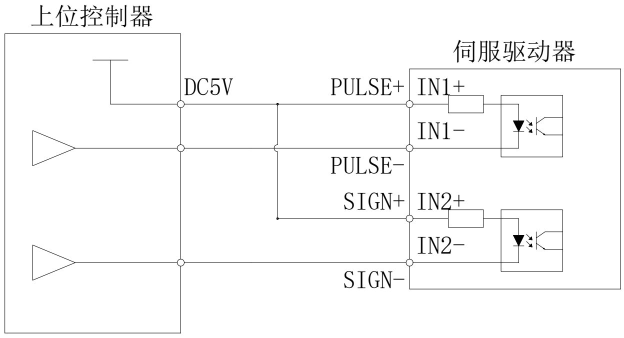


2. External pulse + direction differential signal:

(a) 5V differential input

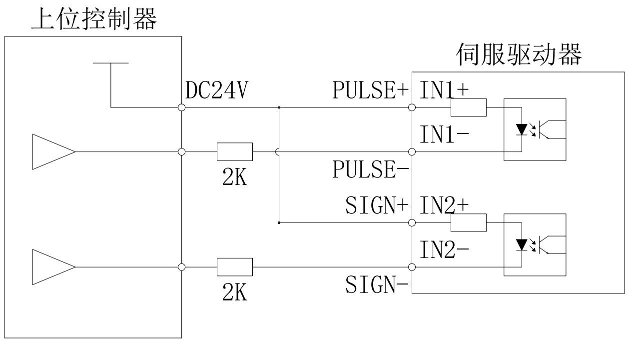


(b) 5V single-ended input

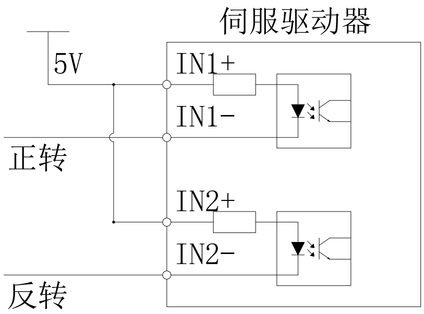


(c) **24V single-ended input**

- **Note: When using 24V input, please connect 2K current limiting resistor externally, otherwise the drive will be damaged.**



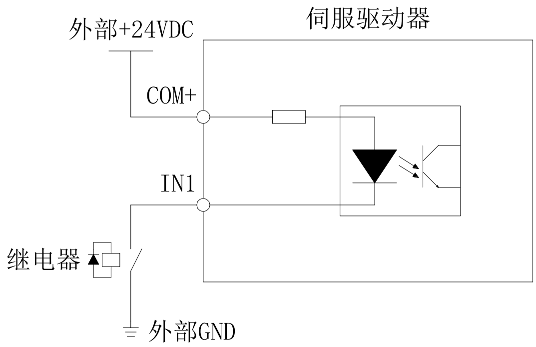
3. Use single-ended connection method, external input signal, such as jog positive/reverse signal:



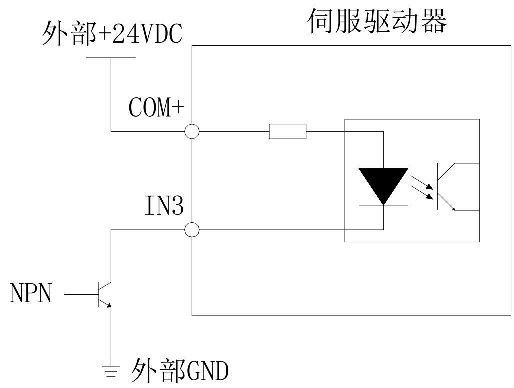
b) IN3 ~ IN6 single-ended input terminal

Taking IN3 as an example, the IN3 to IN6 interface circuits are the same.

1. When the host device is a relay output:

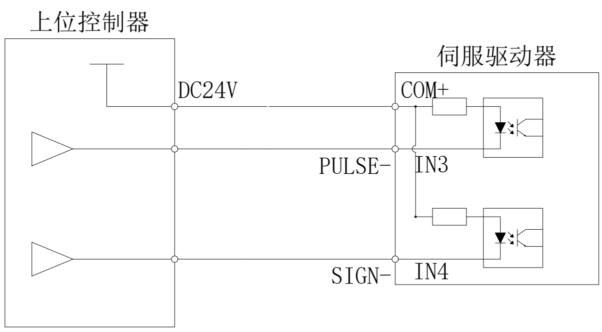


2. When the host device is open collector output:



- **Note: PNP input is not supported.**

3. Pulse + direction signal input using IN3, IN4 terminal



- **If conditions permit, please use IN1 and IN2 as input terminals of pulse + direction signal.**

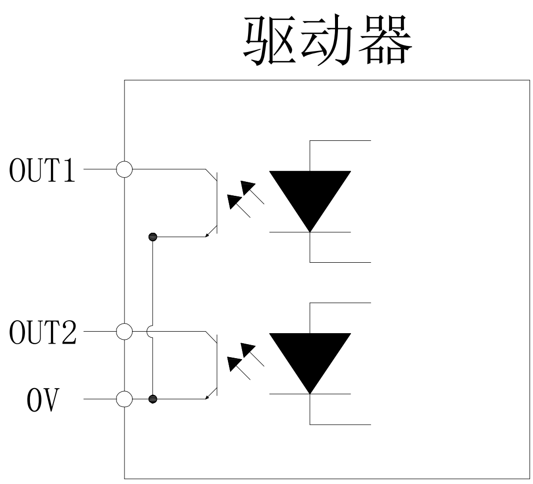
1.4.2 Digital Output Port

The NT86 contains two opto-isolated output signals.

- OUT1's output current capability is up to 30mA.

- OUT2's output current capability is up to 150mA.

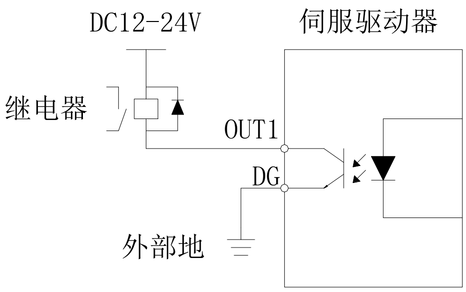
By default, the digital output ports are all normally open. You can use the NTConfigurator debugging software to change the polarity of the output port.



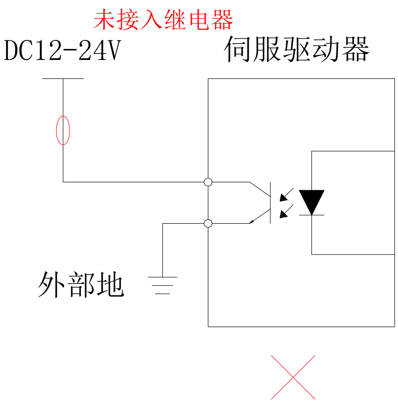
Take OUT1 as an example to illustrate that the OUT1 to OUT2 interface circuits are the same.

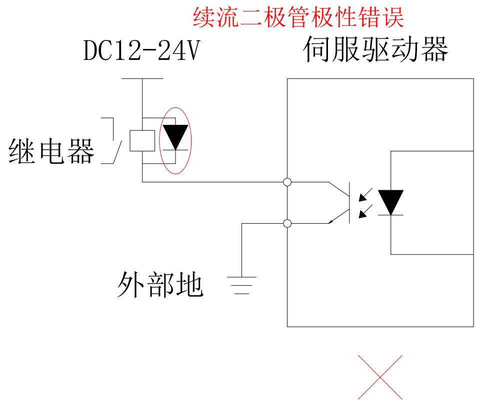
1. When the host device is a relay input:

Correct wiring diagram:

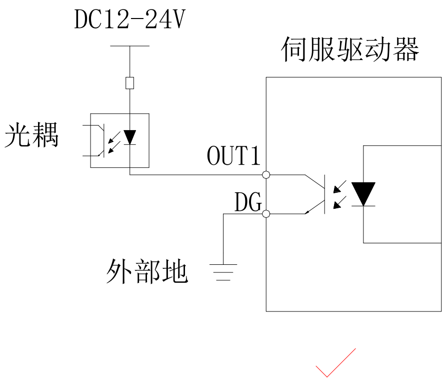


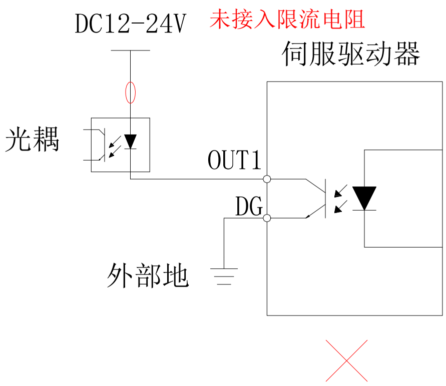
Wrong wiring diagram:





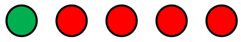
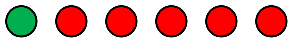
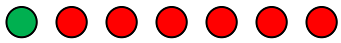
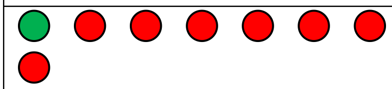
2. When the host device is an optocoupler input:



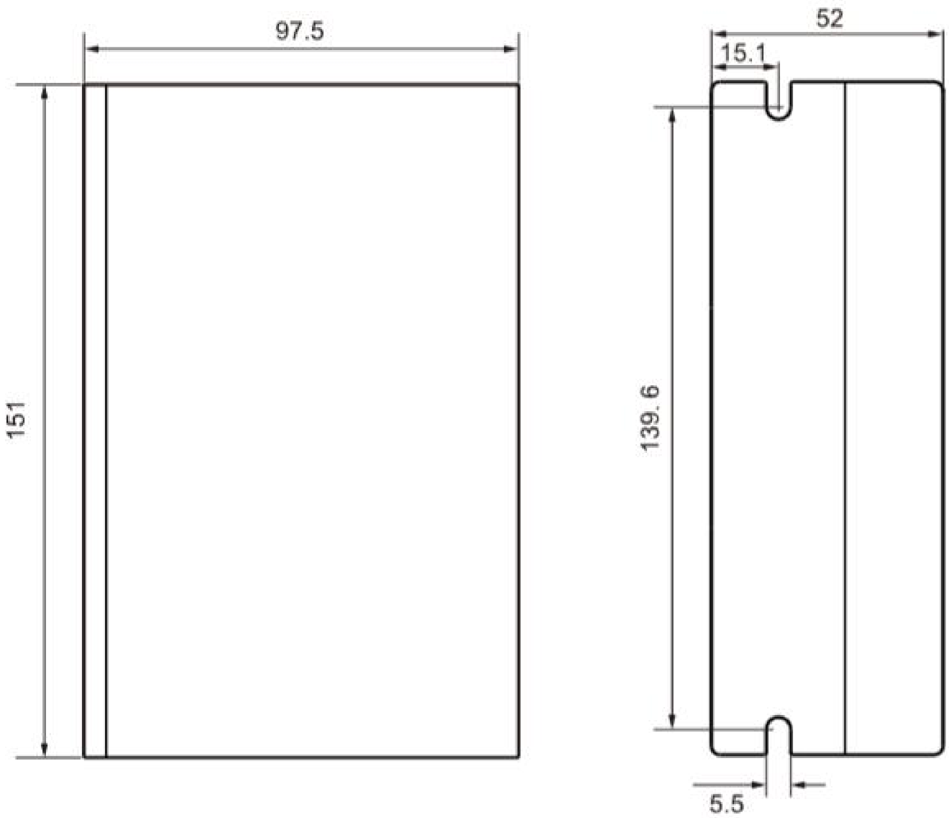


1.5 Alarm code

LED status		Drive status
	Green indicator is on for a long time	Drive not enabled
	Green indicator is flickering	Drive working normally
	One green indicator and one red indicator	Drive overcurrent
	One green indicator and two red indicators	Drive input power overvoltage
	One green indicator and three red indicators	The internal voltage of the drive is wrong

	One green indicator and four red indicators	Tracking error exceeds limits
	One green indicator and five red indicators	Encoder phase error
	One green indicator and six red indicators	Parameter check error
	One green indicator and seven red indicators	Motor phase failure alarm

1.6 Mechanical dimensions



1.7 Accessories

1.7.1 X1 General IO Signal Cable

All 8 signal ports are led out, and shielded cables are used to facilitate customer wiring.

Model	Length (m)	Price (RMB: RMB)

CNT86-250	0.25	8
CNT86-500	0.5	10
CNT86-750	0.75	15
CNT86-1000	1	20

1.7.2 RS-485 extension cable

A network cable that conforms to the CAT6 standard.

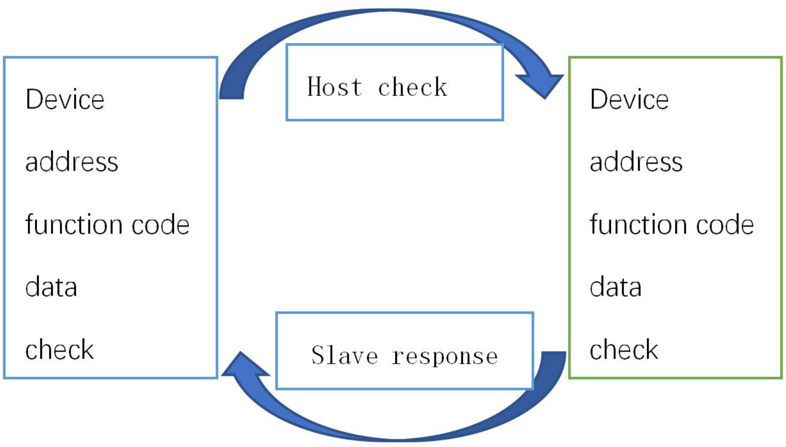
Model	Length (m)	Price (RMB: RMB)
CRJ45-250	0.25	8
CRJ45-500	0.5	10
CRJ45-750	0.75	15
CRJ45-1000	1	20

Chapter 2 Communication Protocol

2.1 Modbus/RTU definition

The Modbus protocol, designed by MODICON, is a bus protocol that allows the primary station to share data with one or more slaves. The data consists of 16-bit registers. The master can read and write a single register or multiple registers. The standard Modbus port on the Modicon controller uses an RS-232 compatible serial interface that defines the connector, the wiring cable,

Signal level, transmission baud rate and parity. Controller communication uses master-slave technology, that is, the host can start data transmission, called query. Other devices (slave) return a response to the query or process the action required by the query. The host device should include the main processor, programmer, and PLC. The slave includes a programmable controller, a servo drive and a stepper drive. Its master-slave query-feedback mechanism is as follows:



2.2 Modbus/RTU message format

Modbus/RTU is a master-slave technology, and the CRC check range is from the device address bit to the data bit; the detailed message format of each function code can be found in the appendix.

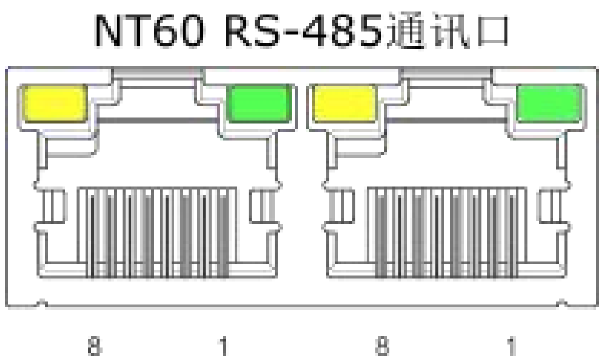
The message frame of Modbus/RTU is as follows:

Address field	function code	Data	CRC check code (2 bytes)

2.3 Modbus/RTU wiring

Modbus/RTU has the same physical layer as standard RS-232 or RS-485, and can configure 1~31 slave addresses; build RS-485 network with topology, usually in the last slave device in parallel with 120 ohm terminal resistance.

NT86/NR60 RS485 network interface definition:



Terminal number	Identifier	colour
1	RS485_A	Orange/white
2	RS485_B	Orange
3	GND	Green/white
4	-	Blue
5	-	Blue/white
6	-	Green
7	-	Brown/white
8	-	Brown

- Note: If you are not using a standard network cable, please refer to the above terminal serial number for correct wiring, instead of wiring according to the color of the network cable!

2.4 Modbus/RTU configuration

You can use the debugging software soft NTConfigurator of the corresponding product through the Rite Electromechanical website (www.szruitech.com) to perform common parameter settings. Users can also use their own host to modify the parameters.

The parameters of Modbus/RTU communication are as follows:

2.4.1 Setting of communication parameters of the master station

1. Baud rate: consistent with the slave station;
2. Data bits: 8 data bits;
3. Stop bit: 1 stop bit;
4. Check digit: There is no check digit.

2.4.2 Configuration of slave communication parameters

1) Slave address

In the same network, each slave has a unique address.

Slave ID	SW1	SW2	SW3	SW4	SW5
Default	ON	ON	ON	ON	ON
1	OFF	ON	ON	ON	ON
2	ON	OFF	ON	ON	ON
3	OFF	OFF	ON	ON	ON
4	ON	ON	OFF	ON	ON
......	......	......	......	......	......
30	ON	OFF	OFF	OFF	OFF
31	OFF	OFF	OFF	OFF	OFF

ON = 0, OFF = 1

Slave address = SW1 + SW2 × 2 + SW3 × 4 + SW4 × 8 + SW5 × 16

2) Baud rate

The master and slave must be set to the same baud rate.

BDR	SW6	SW7
9600	ON	ON
19200	OFF	ON
38400	ON	OFF
115200	OFF	OFF

3) Terminal matching resistor

The end can be selected as appropriate. Usually not needed for short distances.

120 terminating resistor	SW8
Invalid	OFF
Effective	ON

4) Function code supported by Modbus/RTU

The RTELLIGENT driver NT86 currently supports the following Modbus function codes:

a) 0x03: read holding register

b) 0x06: write a single register

c) 0x10: write multiple registers

5) Modbus/RTU register

Register address description

The MODBUS register starts with 0. In the touch screen and PLC, the address of the register is usually expressed as 400x type, starting with 1. and so:

PLC address = MODBUS address +1

Register operation type:

R-read only

W-write only

R/W-readable/writable

type of data:

MODBUS defaults to one register for 16 bits. Two consecutive registers form a 32-bit data.

SHORT - 16bit

LONG - 32bit

2.5 Register Summary

- **Note: The register addresses in the following register summary table are all decimal.**

Table 2-1 Register Summary

Register address	Operation type	type of data	Function Description	Remarks
0	**R**	***SHORT***	Alarm Code	
1	**R**	***SHORT***	Status Code, drive status flag	
2	**R**	***SHORT***	Current input port value	
3	**R**	***SHORT***	Current output port value	
4	**R**	***SHORT***	Universal input port conduction trigger status	
5	**R**	***SHORT***	Universal input port disconnect trigger status	
6	**W**	***SHORT***	Turn-on trigger status clear register	
7	**W**	***SHORT***	Disconnect trigger status clear register	
8	**R**	***SHORT***	In the internal pulse mode, the current absolute position is lower by 16 bits.	Form a LONG type data
9	**R**	***SHORT***	In the internal pulse mode, the current absolute position is 16 bits high.	
10	**R**	***SHORT***	Given speed RPM	
11	**R**	***SHORT***	Bus voltage mV	
12	**R**	***SHORT***	Motor tracking error is 16 bits lower in closed loop mode	Form a LONG type data
13	**R**	***SHORT***	Motor tracking error is 16 bits high in closed loop mode	
14	**R**	***SHORT***	External pulse counter lower 16 bits	Form a LONG type data
15	**R**	***SHORT***	External pulse counter high 16 bits	
16	**W**	***SHORT***	Clear external pulse counter	
17	**R/W**	***SHORT***	Instruction mode: internal command or	

			external pulse	
18	**R/W**	***SHORT***	Motion command with internal command mode and application mode 0	
19	**R**	***SHORT***	Pulse command form when external pulse	
20	**R/W**	***SHORT***	Application mode selection in internal command mode	
21	**R/W**	***SHORT***	Motor type selection: two-phase or three-phase	
22	**R/W**	***SHORT***	Motor control mode selection: open loop, servo mode one, servo mode two	
23	**R/W**	***SHORT***	Reverse the running direction of the motor	
24	**R/W**	**SHORT**	Motor subdivision (pulse number / revolution)	
25	**R/W**	***SHORT***	Operating current (mA)	
26	**R/W**	***SHORT***	Standby current percentage (%)	
27	**R/W**	***SHORT***	Time to enter standby after pulse stop (ms)	
28	**R/W**	***SHORT***	S-shaped acceleration and deceleration time	
29	**R**	***SHORT***	Encoder current position (number of pulses)	
30	**R/W**	***SHORT***	Enable automatic recognition of drive parameters	
31	**R**	***SHORT***	Automatically recognized resistance value mOhm	
32	**R**	***SHORT***	Automatically recognized inductance	

			value mH	
33	**R/W**	***SHORT***	User-set resistance value when canceling automatic recognition	
34	**R/W**	***SHORT***	User-set electrical steel value when canceling automatic recognition	
35	**R/W**	***SHORT***	Motor torque factor, reserved for internal use of the drive	
36	**R/W**	***SHORT***	Current loop proportional gain	
37	**R/W**	***SHORT***	Current loop integral gain	
38	**R/W**	***SHORT***	Current loop phase lead gain	
39	**R/W**	***SHORT***	Current loop step test	
40	**R/W**	***SHORT***	Motor encoder resolution	
41	**R/W**	***SHORT***	Tracking error alarm threshold	
42	**R/W**	***SHORT***	Positioning completion accuracy	
43	**R/W**	***SHORT***	Positioning completion time	
44	**R/W**	***SHORT***	The pulse stops until the time when the detection is completed is completed.	
45	**R/W**	***SHORT***	Maximum current	
46	**R/W**	***SHORT***	Basic current	
47	**R/W**	***SHORT***	Primary speed feedback filter	
48	**R/W**	***SHORT***	Secondary speed feedback filter	
49	**R/W**	***SHORT***	Servo mode - low speed anti-resonance gain	
50	**R/W**	***SHORT***	Servo mode two position loop proportional gain	
51	**R/W**	***SHORT***	Servo mode two position loop integral gain	
52	**R/W**	***SHORT***	Servo mode two speed loop damping 1	
53	**R/W**	***SHORT***	Servo mode two speed loop damping 2	

54	**R/W**	***SHORT***	Servo mode two speed loop feed forward gain	
55	**R/W**	***SHORT***	Servo mode two gravity compensation	
56	**R/W**	***SHORT***	Servo mode two acceleration gain	
57	**R/W**	***SHORT***	Servo mode two acceleration feed forward gain	
58	**R/W**	***SHORT***	Servo mode two speed loop output filter	
59	**R/W**	***SHORT***	Servo mode two acceleration feedforward filter	
60	**R/W**	***SHORT***	Input port 1 setting register	
61	**R/W**	***SHORT***	Input 2 setting register	
62	**R/W**	***SHORT***	Input port 3 setting register	
63	**R/W**	***SHORT***	Input port 4 setting register	
64	**R/W**	***SHORT***	Input port 5 setting register	
65	**R/W**	***SHORT***	Input port 6 setting register	
66	**R/W**	***SHORT***	Output port 1 setting register	
67	**R/W**	***SHORT***	Output port 2 setting register	
68	**R/W**	***SHORT***	Output value setting register for output ports 1, 2 in general-purpose output port mode	
69	**R**	***SHORT***	Input function status	
70	**R/W**	***SHORT***	Point-to-point motion acceleration (r/s^2)	
71	**R/W**	***SHORT***	Point-to-point motion deceleration (r/s^2)	
72	**R/W**	***SHORT***	Point-to-point motion maximum speed (RPM)	
73	**R/W**	***SHORT***	Point-to-point motion travel low 16 bits	Form a LONG type data

			(PUISE)	
74	**R/W**	***SHORT***	Point-to-point motion travel 16 bits high (PUISE)	
75	**R/W**	***SHORT***	Acceleration started during continuous operation (R/S^2)	
76	**R/W**	***SHORT***	Deceleration at deceleration stop during continuous operation (R/S^2)	
77	**R/W**	***SHORT***	Continuous running speed (RPM)	
78	**R/W**	***SHORT***	Deceleration during emergency stop	
79	**R/W**	***SHORT***	Zero return mode selection	
80	**R/W**	***SHORT***	Zero return high speed	
81	**R/W**	***SHORT***	Zero return low speed	
82	**R/W**	***SHORT***	Zero return acceleration	
83	**R/W**	***SHORT***	Position offset after completion of zero return	
84	**R/W**	***SHORT***	Position mode selection: incremental motion and absolute motion	
85	**R/W**	***SHORT***	Internal instruction counter clear	
88	**R/W**	***SHORT***	Incomplete alarm is invalid	
89	**R/W**	***SHORT***	Servo mode-integral gain	
90	**R/W**	***SHORT***	Writing a 1 will save the current parameter and then automatically clear it.	
91	**R/W**	***SHORT***	Writing a 1 will restore the factory settings and then automatically clear	
92	**R**	***SHORT***	Vendor reserved, do not write any value in this register	
93	**R**	***SHORT***	Drive ID	
94	**R**	***SHORT***	Drive version	

95	**R**	***SHORT***	Non-label	
100	**R/W**	***SHORT***	IO switching effective time in speedometer and position table mode	
101	**R/W**	***SHORT***	Current step test current (mA)	
102	**R/W**	***SHORT***	Output port 3 setting register	
103	**R/W**	***SHORT***	Output port 4 setting register	
104	**R**	***SHORT***	Output port mark	
105	**R/W**	***SHORT***	Internal speed 0	
106	**R/W**	***SHORT***	Internal speed 1	
107	**R/W**	***SHORT***	Internal speed 2	
108	**R/W**	***SHORT***	Internal speed 3	
109	**R/W**	***SHORT***	Internal speed 4	
110	**R/W**	***SHORT***	Internal speed 5	
111	**R/W**	***SHORT***	Internal speed 6	
112	**R/W**	***SHORT***	Internal speed 7	
113	**R/W**	***SHORT***	Internal speed 8	
114	**R/W**	***SHORT***	Internal speed 9	
115	**R/W**	***SHORT***	Internal speed 10	
116	**R/W**	***SHORT***	Internal speed 11	
117	**R/W**	***SHORT***	Internal speed 12	
118	**R/W**	***SHORT***	Internal speed 13	
119	**R/W**	***SHORT***	Internal speed 14	
120	**R/W**	***SHORT***	Internal speed 15	
121	**R/W**	***SHORT***	Currently triggered position table	
122	**R/W**	***SHORT***	Default parameter ID number	
125	**R/W**	***SHORT***	Internal position 0 lower 16 bits	Form a LONG type data
126	**R/W**	***SHORT***	Internal position 0 high 16 bits	
127	**R/W**	***SHORT***	Internal position 1 low 16 bits	Form a LONG type data
128	**R/W**	***SHORT***	Internal position 1 high 16	

129	**R/W**	***SHORT***	Internal position 2 low 16 bits	Form a LONG type data
130	**R/W**	***SHORT***	Internal position 2 high 16 bits	
131	**R/W**	***SHORT***	Internal position 3 low 16 bits	Form a LONG type data
132	**R/W**	***SHORT***	Internal position 3 high 16 bits	
133	**R/W**	***SHORT***	Internal position 4 low 16 bits	Form a LONG type data
134	**R/W**	***SHORT***	Internal position 4 high 16 bits	
135	**R/W**	***SHORT***	Internal position 5 low 16	Form a LONG type data
136	**R/W**	***SHORT***	Internal position 5 high 16 bits	
137	**R/W**	***SHORT***	Internal position 6 low 16 bits	Form a LONG type data
138	**R/W**	***SHORT***	Internal position 6 high 16 bits	
139	**R/W**	***SHORT***	Internal position 7 lower 16 bits	Form a LONG type data
140	**R/W**	***SHORT***	Internal position 7 high 16	
141	**R/W**	***SHORT***	Internal position 8 low 16 bits	Form a LONG type data
142	**R/W**	***SHORT***	Internal position 8 high 16 bits	
143	**R/W**	***SHORT***	Internal position 9 low 16	Form a LONG type data
144	**R/W**	***SHORT***	Internal position 9 high 16	
145	**R/W**	***SHORT***	Internal position 10 low 16 bits	Form a LONG type data
146	**R/W**	***SHORT***	Internal position 10 high 16 bits	
147	**R/W**	***SHORT***	Internal position 11 lower 16 bits	Form a LONG type data
148	**R/W**	***SHORT***	Internal position 11 high 16	
149	**R/W**	***SHORT***	Internal position 12 is lower 16 bits	Form a LONG type data
150	**R/W**	***SHORT***	Internal position 12 high 16 bits	
151	**R/W**	***SHORT***	Internal position 13 is lower 16 bits	Form a LONG type data
152	**R/W**	***SHORT***	Internal position 13 high 16	
153	**R/W**	***SHORT***	Internal position 14 is lower 16 bits	Form a LONG type data
154	**R/W**	***SHORT***	Internal position 14 high 16	
155	**R/W**	***SHORT***	Internal position 15 low 16	构成一个 ***LON*** 型数据
156	**R/W**	***SHORT***	Internal position 15 high 16	
157	**R/W**	***SHORT***	Torque mode speed loop proportional	

			gain	
158	**R/W**	***SHORT***	Torque mode speed loop integral gain	
214	**R/W**	***SHORT***	The 3.3V voltage input corresponds to the lower 16 bits of the pulse command.	Form a LONG type data
215	**R/W**	***SHORT***	3.3V voltage input corresponding pulse command high 16 bits	
216	**R**	***SHORT***	The current input voltage corresponds to the lower 16 bits of the position command	Form a LONG type data
217	**R**	***SHORT***	The position command corresponding to the current input voltage is 16 bits high.	
218	**R/W**	***SHORT***	Set the command error range without analog position adjustment	
221	**R/W**	***SHORT***	Multi-segment position operation mode setting	
222	**R/W**	***SHORT***	Multi-segment position shift end segment number setting	
223	**R/W**	***SHORT***	Multi-segment position operation waiting time unit setting	
224	**R/W**	***SHORT***	1st stage displacement maximum running speed	
225	**R/W**	***SHORT***	Stage 1 displacement acceleration and deceleration	
226	**R/W**	***SHORT***	Waiting time after the completion of the first stage displacement	
227	**R/W**	***SHORT***	2nd stage displacement maximum running speed	

228	**R/W**	***SHORT***	Stage 2 displacement acceleration and deceleration	
229	**R/W**	***SHORT***	Waiting time after the completion of the second stage displacement	
230	**R/W**	***SHORT***	The third stage displacement maximum running speed	
231	**R/W**	***SHORT***	Stage 3 displacement acceleration and deceleration	
232	**R/W**	***SHORT***	Waiting time after the completion of the third stage displacement	
233	**R/W**	***SHORT***	Stage 4 displacement maximum running speed	
234	**R/W**	***SHORT***	Stage 4 displacement acceleration and deceleration	
235	**R/W**	***SHORT***	Waiting time after the completion of the fourth stage displacement	
236	**R/W**	***SHORT***	5th stage displacement maximum running speed	
237	**R/W**	***SHORT***	Stage 5 displacement acceleration and deceleration	
238	**R/W**	***SHORT***	Waiting time after the 5th segment is completed	
239	**R/W**	***SHORT***	Stage 6 displacement maximum running speed	
240	**R/W**	***SHORT***	Stage 6 displacement acceleration and deceleration	
241	**R/W**	***SHORT***	Waiting time after the completion of the sixth stage displacement	
242	**R/W**	***SHORT***	7th stage displacement maximum	

			running speed	
243	**R/W**	***SHORT***	Stage 7 displacement acceleration and deceleration	
244	**R/W**	***SHORT***	Waiting time after the completion of the 7th segment displacement	
245	**R/W**	***SHORT***	Stage 8 displacement maximum running speed	
246	**R/W**	***SHORT***	Stage 8 displacement acceleration and deceleration	
247	**R/W**	***SHORT***	Waiting time after the 8th segment is completed	
248	**R/W**	***SHORT***	The maximum speed of the 9th displacement	
249	**R/W**	***SHORT***	Stage 9 displacement acceleration and deceleration	
250	**R/W**	***SHORT***	Waiting time after the 9th segment is completed	
251	**R/W**	***SHORT***	Stage 10 displacement maximum running speed	
252	**R/W**	***SHORT***	Stage 10 displacement acceleration and deceleration	
253	**R/W**	***SHORT***	Waiting time after the completion of the 10th displacement	
254	**R/W**	***SHORT***	Stage 11 displacement maximum running speed	
255	**R/W**	***SHORT***	Stage 11 displacement acceleration and deceleration	
256	**R/W**	***SHORT***	Waiting time after the completion of the 11th displacement	

257	**R/W**	***SHORT***	Stage 12 displacement maximum running speed	
258	**R/W**	***SHORT***	Stage 12 displacement acceleration and deceleration	
259	**R/W**	***SHORT***	Waiting time after the completion of the 12th displacement	
260	**R/W**	***SHORT***	Stage 13 displacement maximum running speed	
261	**R/W**	***SHORT***	Stage 13 displacement acceleration and deceleration	
262	**R/W**	***SHORT***	Waiting time after the completion of the 13th displacement	
263	**R/W**	***SHORT***	Stage 14 displacement maximum running speed	
264	**R/W**	***SHORT***	Stage 14 displacement acceleration and deceleration	
265	**R/W**	***SHORT***	Waiting time after the completion of the 14th displacement	
266	**R/W**	***SHORT***	The maximum speed of the 15th displacement	
267	**R/W**	***SHORT***	Stage 15 displacement acceleration and deceleration	
268	**R/W**	***SHORT***	Waiting time after the completion of the 15th displacement	
269	**R/W**	***SHORT***	Stage 16 displacement maximum running speed	
270	**R/W**	***SHORT***	Stage 16 displacement acceleration and deceleration	

271	**R/W**	***SHORT***	Waiting time after the completion of the 16th displacement	
272	**R/W**	***SHORT***	Analog input offset	
273	**R/W**	***SHORT***	Analog input low pass filter cutoff frequency	
274	**R/W**	***SHORT***	Analog input deadband	
275	**R/W**	***SHORT***	Analog input zero drift	
276	**R/W**	***SHORT***	3.3V voltage input corresponding speed command	
277	**R**	***SHORT***	DSP current sampling voltage value	
278	**R**	***SHORT***	Input voltage value after zero drift, dead zone, and offset processing	
279	**R**	***SHORT***	Current speed input corresponding speed	
280	**R/W**	***SHORT***	Modbus bus error counter	
281	**R/W**	***SHORT***	Modbus CRC error counter	
282	**R/W**	***SHORT***	Modbus Receive Byte Count Error Counter	
287	**R/W**	***SHORT***	Origin return enable control	
288	**R/W**	***SHORT***	Origin return mode	
289	**R/W**	***SHORT***	High speed search for the origin signal speed	
290	**R/W**	***SHORT***	Searching for the speed of the origin signal at low speed	
291	**R/W**	***SHORT***	Search for the acceleration and deceleration of the origin signal	
292	**R**	***SHORT***	Reserved	
293	**R/W**	***SHORT***	Mechanical origin offset is 16 bits lower	

294	**R/W**	***SHORT***	Mechanical origin offset is 16 bits high	
295	**R/W**	***SHORT***	Mechanical origin offset processing	
296	**R/W**	***SHORT***	Collision back to origin detection time	
297	**R/W**	***SHORT***	Collision back to origin speed judgment threshold	
298	**R/W**	***SHORT***	Collision back to origin torque limit	

2.6 Register Details

2.6.1 Driver Flag Register [0-1]

2.6.1.1 Alarm Mark Register [0]

All alarm signs of the driver are defined. MODBUS address: 0

15				11	10	9	8
Reserve							ECDE1
		R-0				R-0	

7	6	5	4	3	2	1	0
POSE	MPE	MEM	OT	UV	OV	OC	IVE
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0

BIT	Name	Description
9～15	Reserve	Read always returns 0
8	ECDE1	encoder failed 0: Encoder signal is normal 1: Abnormal encoder signal
7	POSE	Tracking error alarm 0: no tracking error alarm 1: Tracking error alarm occurs, the motor can not follow the encoder normally. The possible impacts are as follows: Alarm threshold for position overshoot Wiring of encoder Wiring of motor

		Whether the setting of speed, acceleration and other parameters is reasonable
6	MPE	Phase-missing alarm of motor 0: Phase-absent alarm 1: Phase absence alarm occurs, the driver can not detect the current of the motor winding normally. Need to test motor wiring, motor type
5	MEM	Error in parameter checking 0: Correct parameter verification 1: Error in parameter checking.
4	OT	Overtemperature alarm sign 0: Driver temperature is normal 1: Driver internal device temperature is too high
3	UV	Undervoltage alarm sign 0:No under-voltage alarm 1: Driver undervoltage
2	OV	Overvoltage alarm sign 0: no overvoltage alarm 1: Driver overvoltage, need to do the following detection: Check the input power supply Check the Pump-up Voltage when the Motor Decelerates
1	OC	Overcurrent alarm sign 0: no overcurrent alarm 1: Overcurrent alarm happened to the driver. Possible reasons are as follows: Short Circuit Occurs in Motor Winding The excessive current set by the driver causes the motor to burn down Damage to internal components of driver
0	IVE	Internal Voltage Error Alarm Mark 0: No internal voltage error 1: Internal voltage error, usually caused by damage to internal components of the driver

2.6.1.2 Driver Status Register [1]

Some state flags inside the drive are defined. MODBUS address: 1

BIT	Name	Description
8～15	Reserve	Read always returns 0
11	TC	Torque arrival state 0: The torque has not reached the set value 1: torque reaches the set value
10	POW	Power status 0: The drive is not powered 1: Drive power supply
9	NL	Negative limit valid status 0: not in the negative limit position 1: in the negative limit position
8	PL	Positive limit validity 0: not in the positive position 1: in the positive position

15				11	10	9	8
Reserve				TC	POW	NL	PL

R-0

7	6	5	4	3	2	1	0
CLAMP	ARRSPD	RDY	HOME	MOV	INPOS	ALM	ENA
R-0	R-0	R-0	R-1	R-0	R-0	R-0	R-1

7	CLAMP	Motor mechanical brake state 0: Brake is not open, mechanically locks the motor shaft 1: The brake is open and the motor can run
6	ARRSPD	Whether the motor runs to the set speed 0: Speed has not arrived

		1: speed has arrived In the internal pulse command mode, it is used to indicate whether the motor has reached the set speed.
5	RDY	Drive ready flag 0: not ready 1: ready Normally the drive is in the ready state when it is enabled. However, it takes 100ms for the motor to be in the ready state during the transition from the enable to the enable. Automatic identification of the parameters as above and current step testing will result in the motor not being ready.
4	HOME	Zero mark 0: zero return is not completed 1: zero return has been completed
3	MOV	Motor movement sign 0: Motor stop state 1: The motor is running When the motor is running, it cannot respond to new motion commands and can only respond to stop commands.
2	INPOS	Motor positioning completion flag in closed loop mode 0: Positioning is not completed 1: Positioning completed
1	ALM	Drive alarm flag 0: The drive has no alarm 1: The drive has an alarm. Please check the status of the register REG_ALMCODE (address 0).
0	ENA	Drive enable flag 0: Drive is not enabled 1: The drive is enabled The default drive is powered on.

2.6.2 I/O state registers [2-7]

2.6.2.1 Input Port Value Register [2]

Used to indicate the value of the current input port. Since the input port is photoelectric isolation, in order to facilitate understanding, the state of the input port is represented by whether the optocoupler is on or off. MODBUS address:2

15 … 8
Reserve
R-0

7 6	5	4	3	2	1	0
Reserve	IN6	IN5	IN4	IN3	IN2	IN1
R-0	R-0	R-0	R-0	R-0	R-0	R-0

BIT	Name	Description
6～15	Reserve	Read always returns 0
5	IN6	Input port IN6 level status 0: Input port 6 is not conducting 1: Input port 6 is turned on
4	IN5	Input port IN5 level status 0: Input port 5 is not conducting 1: Input port 5 is turned on
3	IN4	Input port IN4 level status 0: Input port 4 is not conducting 1: Input port 14 is turned on
2	IN3	Input port IN3 level status 0: Input port 3 is not conducting 1: Input port 3 is turned on
1	IN2	Input port IN2 level status 0: Input port 2 is not conducting 1: Input port 2 is turned on
0	IN1	Input port IN1 level status 0: Input port 1 is not conducting 1: Input port 1 is turned on

2.6.2.2 Value of the current output port [3]

Output port value register. MODBUS address: 3

15					8
Reserve					
R-0					

7 — 4	3	2	1	0
Reserve	OUT4	OUT3	OUT2	OUT1
R-0	R-0	R-0	R-0	R-0

BIT	Name	Description
4～15	Reserve	Read always returns 0
3	OUT4	Level status of output port 4 (used by other products) 0: Output port 4 is not conducting 1: Output port 4 is turned on
2	OUT3	Level status of output port 3 (used by other products) 0: Output port 3 is not conducting 1: Output port 3 is turned on
1	OUT2	Output port 2 level status 0: Output port 2 is not conducting 1: Output port 2 is turned on
0	OUT1	Output port 1 level status 0: Output port 1 is not conducting 1: Output port 1 is turned on

2.6.2.3 Input Port Conduction Edge Latch Register [4]

Each time the port changes from the off state to the on state, the driver will latch this change edge. MODBUS address: 4

15 8

Reserve
R-0

7	6	5	4	3	2	1	0
Reserve		IN6	IN5	IN4	IN3	IN2	IN1
R-0		R-0	R-0	R-0	R-0	R-0	R-0

BIT	Name	Description
6～15	Reserve	Read always returns 0
5	IN6	Input port IN6 conduction edge latch flag 0: No conduction has occurred on input port 6. 1: Input port 6 has a conduction edge
4	IN5	Input port IN5 turns on the latch flag 0: No conduction edge has occurred on input port 5. 1: Input port 5 has a conduction edge
3	IN4	Input port IN4 conduction edge latch flag 0: No conduction has occurred on input port 4. 1: Input port 4 has a conduction edge
2	IN3	Input port IN3 conduction edge latch flag 0: No conduction edge has occurred on input port 3. 1: Input port 3 has a conduction edge
1	IN2	Input port IN2 conduction edge latch flag 0: No conduction edge has occurred on input port 2. 1: Input port 2 has a conduction edge
0	IN1	Input port IN1 turn-on edge latch flag 0: No conduction has occurred on input port 1. 1: Input port 1 has a conduction edge

2.6.2.4 Input Port Shutdown Edge Latch Register [5]

Each time the port changes from on to off, the driver will latch this change edge.

MODBUS address: 5

BIT	Name	Description
6～15	Reserve	Read always returns 0
5	IN6	Input port IN6 shutdown edge latch flag 0: Input port 6 has not experienced a shutdown edge 1: Input port 6 has a shutdown edge
4	IN5	Input port IN5 shutdown edge latch flag 0: No shutdown edge has occurred on input port 5. 1: Input port 5 has a shutdown edge
3	IN4	Input port IN4 shutdown edge latch flag 0: Input port 4 has not had a shutdown edge 1: Input port 4 has a turn-off edge

15 … 8
Reserve
R-0

7 … 6	5	4	3	2	1	0
Reserve	IN6	IN5	IN4	IN3	IN2	IN1
R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

BIT	Name	Description
2	IN3	Input port IN3 shutdown edge latch flag 0: Input port 3 has not experienced a shutdown edge 1: Input port 3 has a shutdown edge
1	IN2	Input port IN2 turn-off edge latch flag 0: Input port 2 has not experienced a shutdown edge 1: Input port 2 has a shutdown edge
0	IN1	Input port IN1 shutdown edge latch flag 0: Input port 1 has not experienced a shutdown edge 1: Input port 1 has a shutdown edge

2.6.2.5 Input Port On Edge Clear Register [6]

Used to clear the latched on-edge flag. MODBUS address: 6

BIT	Name	Description
6～15	Reserve	Read always returns 0
5	IN6	Clear the on-latch state flag of IN6 0: no effect 1: Clear the on-edge latch flag of the IN6 port
4	IN5	Clear the on-latch state flag of IN5 0: no effect 1: Clear the on-edge latch flag of the IN5 port
3	IN4	Clear the on-latch state flag of IN4 0: no effect 1: Clear the on-edge latch flag of the IN4 port
2	IN3	Clear the on-latch state flag of IN3 0: no effect 1: Clear the on-edge latch flag of the IN3 por
1	IN2	Clear the on-latch state flag of IN2 0: no effect 1: Clear the on-edge latch flag of the IN2 port
0	IN1	Clear the on-latch state flag of IN1 0: no effect 1: Clear the on-edge latch flag of the IN1 port

15–8
Reserve
R-0

7–6	5	4	3	2	1	0
Reserve	IN6	IN5	IN4	IN3	IN2	IN1
R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

2.6.2.6 Input Port Shutdown Edge Clear Register [7]

A shutdown edge flag used to clear the latch. MODBUS address: 7

BIT	Name	Description
6～15	Reserve	Read always returns 0
5	IN6	Clear the off-edge latch status flag of IN6 0: no effect 1: Clear the shutdown edge latch flag of IN6 port
4	IN5	Clear the shutdown edge latch status flag of IN5 0: no effect 1: Clear the shutdown edge latch flag of the IN5 port
3	IN4	Clear the off-edge latch status flag of IN4 0: no effect 1: Clear the shutdown edge latch flag of IN4 port
2	IN3	Clear the shutdown edge latch status flag of IN3 0: no effect 1: Clear the shutdown edge latch flag of the IN3 port
1	IN2	Clear the off-edge latch status flag of IN2 0: no effect 1: Clear the shutdown edge latch flag of the IN2 port
0	IN1	Clear the shutdown edge latch status flag of IN1 0: no effect 1: Clear the shutdown edge latch flag of IN1 port

15 ~ 8
Reserve
R-0

7	6	5	4	3	2	1	0
Reserve		IN6	IN5	IN4	IN3	IN2	IN1
R-0		R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0

2.6.3 Current position and speed related registers of the motor [8~16]

MODBUS address	Attributes	Defaults	Range	description
8	R	0	[0,65535]	In the internal pulse mode, the current absolute position is lower by 16 bits.
9	R	0	[0,65535]	In the internal pulse mode, the current absolute position is 16 bits high.
10	R	0	[-3000,3000]	Current command speed. Signed 16-bit data in RPM
11	R	-	[0,100]	Current bus voltage value in mV
12	R	0	[0,65535]	Motor tracking error is 16 bits lower in closed loop mode Unit: encoder resolution
13	R	0	[0,65535]	Motor tracking error is 16 bits in closed loop mode
14	R	0	[0,65535]	External pulse counter lower 16 bits
15	R	0	[0,65535]	External pulse counter high 16 bits
16	R/W	0	[0,1]	Clear external pulse counter Write 0 has no effect, read always returns 0 Writing a 1 clears the external pulse counter and the values of registers 14, 15 become zero. This register will then change to 0.

2.6.4 Drive Control Mode Settings [17~23]

MODBUS address	Attributes	Defaults	Range	description
17	R/W	0	[0,1]	Command mode setting register, set the source of the pulse command of the driver

				0: internal pulse command 1: external pulse command
18	R/W	0	[0,6]	Control instruction in internal pulse mode 0: Waiting state. The drive receives any control commands and will resume the bit wait state after processing by the drive. So reading this register always returns 0. 1: fixed length forward. In the relative position mode, the motor runs in the forward direction according to the 70-74 register parameters. In the absolute position mode, the operating state is determined according to the current position and the absolute position set by 70 to 74. 2: Fixed length reversal. In the relative position mode, the motor runs in reverse according to the 70-74 register parameters. In the absolute position mode, the operating state is determined according to the current position and the absolute position set by 70 to 74. 3: Speed mode, continuous forward rotation. The motor performs forward acceleration operation according to the 75 and 77 registers. 4: Speed mode, continuous reversal. The motor performs reverse acceleration operation according to the 75 and 77 registers. 5: Emergency stop.

				The motor decelerates and stops according to the 78 register 6: Deceleration stops. Position mode, the motor decelerates and stops according to the 71 register Speed mode, the motor decelerates and stops according to the 76 register Other: No effect. This register only works when the internal pulse mode register 20 value is 0.
19	R/W	0	[0,2]	External pulse command mode setting register 0: IN1 is the pulse input port and IN2 is the direction input port. 1: IN1 is the positive pulse input port, and IN2 is the reverse pulse input port. 2: IN1 is the quadrature encoder A phase input port, and IN2 is the quadrature encoder B phase input port. Other: invalid Note Mode 2 here, although the driver receives the quadrature encoder signal, but the driver only follows it, which is a form of instruction. It is not the position feedback signal of the stepper motor itself. This function can be used to follow the encoder signal output by other devices such as servo drives.
20	R/W	0	[0,5]	Preset application selection in internal pulse mode 0: command in response to 18 registers 1: Reserved, do not use 2: preset IO control mode 1: start and stop + direction

				3: preset IO control mode 2: forward + reverse 4: preset IO control mode 3: internal speedometer 5: preset IO control mode 4: internal position table 6: preset IO control mode five: step position 7: Customization 1 8: Customization 2 9: Customization 3 10: Customization 4 11: Customization 5 12: Customization 6 13: Customization 7 14: Customization 8 15: Customization 9 16: Customization 10 17: Customization 11 18: Customization 12 19: Customization 13 20: Customization 14 21: Analog speed (customized 15) 22: Analog position (customized 16)
21	R/W	0	[0,1]	Motor type setting register 0: two-phase stepper motor 1: three-phase stepper motor
22	R/W	0	[0,2]	Motor operating mode setting register 0: open loop operation 1: Servo mode one 2: Servo mode 2
23	R/W	0	[0,1]	Motor direction inversion setting register 0: default running direction 1: Reverse the running direction of the motor

2.6.5 Open loop operation parameter setting [24~29]

MODBUS address	Attributes	Defaults	Range	description
24	R/W	4000	[200,65535]	Segmentation settings Set the breakdown of the drive runtime
25	R/W	3000	[0,6000]	Open loop operating current The sine peak of the drive when it is open loop. Unit: mA
26	R/W	50	[0,100]	Standby current percentage Set the open-loop operating mode of the drive as a percentage of the current flowing into standby mode relative to the operating current. unit:%
27	R/W	500	[10,65535]	Standby time setting Set the time when the drive enters the standby state after the pulse stops for a certain period of time when the drive is open loop. Unit: ms
28	R/W	128	[1,512]	Pulse command filter For smoothing pulse commands (both internal and external), filter time = set value * 50us
29	R	-	-	Current position of the encoder (number of pulses)

2.6.6 Motor and current loop parameters [30~39]

MODBUS address	Attributes	Defaults	Range	description
30	R/W	0	[0,1]	Automatic PI enable function The drive has built-in parameter identification and gain optimization algorithms. Usually good results can be achieved. If the customer needs optimization, this feature can be disabled. 0: Do not use automatic PI function 1: Use automatic PI function
31	R	-	[100,65535	Automatically recognized resistance value

			]	Read the motor winding resistance value automatically recognized by the drive. Unit: mOhm
32	R	-	[1,65535]	Automatically recognized inductance value Read the motor winding inductance value automatically recognized by the drive. Unit: mH
33	R/W	1000	[100,10000]	User-set resistance value When the automatic PI function is canceled, the resistance value set by the user takes effect. Unit: mOhm
34	R/W	1	[1,10]	User-set inductance value When the automatic PI function is canceled, the inductance value set by the user is valid. Unit: mH
35	R/W	200	[0,1000]	Motor torque constant The parameter is valid only when the motor control mode is servo mode 2
36	R/W	1000	[200,10000]	Current loop proportional gain KP in the current loop PI algorithm. When the automatic PI function is enabled, ILOOPKP is automatically generated. When the automatic PI function is not enabled, the user can modify ILOOPKP.
37	R/W	200	[0,2000]	Current loop integral gain KI in the current loop PI algorithm. When the automatic PI function is enabled, ILOOPKI is automatically generated. When the automatic PI function is not enabled, the user can modify ILOOPKI.
38	R/W	256	[0,1024]	KC in the current loop PI algorithm.
39	R/W	0	[0,1]	Current step test Write 0 has no effect, read always returns 0 Writing a 1 will initiate a current loop step test. At this point the current in the motor winding will first be 0 and then increase to 1000 mA. Users can view the step response through NTConfigurater, manually adjust ILOOPKP and ILOOPKI to optimize motor response.

2.6.7 Closed loop control motor parameters [40~48]

MODBUS address	Attributes	Defaults	Range	description
40	R/W	4000	[256,65535]	Encoder feedback resolution The driver is capable of receiving quadrature encoder input signals and performing 4x processing. Encoder resolution = number of encoder lines X 4
41	R/W	2000	[100,65535]	Tracking error alarm threshold The alarm threshold is in encoder resolution.
42	R/W	10	[1,65535]	Positioning completion accuracy In encoder resolution.
43	R/W	50	[1,65535]	Positioning completion duration After setting the motor to the completion accuracy, the duration, duration = set value X 50us
44	R/W	100	[1,65535]	The time when the positioning is completed and the detection is started. After setting the drive to stop receiving pulses, the set time is elapsed, and then it is judged whether the positioning is completed. Set time = set value X 50us
45	R/W	4000	[0,5000]	Maximum current of closed loop control Set the maximum allowable current when the drive is closed-loop, sinusoidal peak, unit: mA
46	R/W	50	[0,100]	Base current percentage for closed loop control
47	R/W	200	[10,5000]	Primary speed filter, unit: Hz
48	R/W	600	[10,5000]	Secondary speed filtering, unit: Hz

2.6.8 Closed loop servo parameters [49~59]

MODBUS address	Attributes	Defaults	Range	description
49	R/W	0	[0,500]	Servo mode - low speed anti-resonance gain
50	R/W	3000	[0,65535]	Servo mode two position loop proportional gain

BIT	Name			Description
51	R/W	1000	[0,65535]	Servo mode two position loop integral gain
52	R/W	0	[0,65535]	Servo mode two speed loop damping 1
53	R/W	800	[0,65535]	Servo mode two speed loop damping 2
54	R/W	600	[0,65535]	Servo mode two speed loop feed forward gain
55	R/W	512	[0,1024]	Servo mode two gravity compensation
56	R/W	0	[0,65535]	Servo mode two acceleration gain
57	R/W	0	[0,65535]	Servo mode two acceleration feed forward gain
58	R/W	5000	[10,5000]	Servo mode two speed loop output filter
59	R/W	2000	[10,5000]	Servo mode two acceleration feedforward filter

2.6.9 Input/Output Setting Registers [60~69], [102~104]

15 ~ 8
Reserve
R-0

7 ~ 6	5	~ 0
Reserve	GPINPOLARITY	GPINPUTFUNC
R-0	R/W-0	R/W-0

2.6.9.1 Input port setting register [60~65]

The drive contains 6 input ports, each of which is set in the same way.

6～15	Reserve	Read always returns 0
5	GPINPOLARITY	Active level of the input port 0: normally closed 1: Normally open (default)
0～4	GPINPUTFUNC	Input port function selection 0: pulse input 1: direction input 2: Quadrature encoder A phase input 3: Quadrature encoder B phase input 4: Motor offline 5: Clear the fault 6: Emergency stop 7: Jog forward/start/stop 8: Jog reverse / direction 9: Positive limit input 10: Reverse limit input 11: Zero signal 12: Start zero return 13: Reverse the running direction of the motor 14: Multi-speed control 0 15: Multi-speed control 1 16: multi-speed control 2 17: Multi-speed control 3 18: Multi-segment position control 0 19: Multi-segment position control 1 20: Multi-segment position control 2 21: Multi-segment position control 3 22: USER1 23: USER2 24: USER3 25: USER4 26: USER5 27: USER6 28: USER7 29: USER8 30: USER9

		31: USER10 Other: The input port has no effect, only the ordinary input port

MODBUS address	Attributes	Defaults	Range	description
60	R/W	0	[0,31]	Input port 1 setting register
61	R/W	1	[0,31]	Input 2 setting register
62	R/W	4	[0,31]	Input port 3 setting register
63	R/W	7	[0,31]	Input port 4 setting register
64	R/W	12	[0,31]	Input port 5 setting register
65	R/W	11	[0,31]	Input port 6 setting register

2.6.9.2 Output port setting register [66~69]

The drive contains two output ports, each of which is set in the same way

15 ... 8
Reserve
R-0

7 6 5	4	3 ... 0
Reserve	OUT_POLARITY	GPOUTPUTFUNC
R-0	R/W-0	R/W-0

BIT	Name	Description
5～15	Reserve	Read always returns 0
4	OUT_POLARITY	Output port polarity 0: normally closed 1: Normally open (default)

0～3	GPOUTPUTFUNC	Output port function selection 0: normal output, user control 1: Alarm output, OUT0 default value 2: Brake signal output 3: In-position signal output 4: Speed reaches output, OUT1 default value 5: Return to zero to complete the output 6: The drive is ready for output 7: Motor stop status output 8: Positive limit output 9: Negative limit output 10: Power indicator output 11: Torque reaches the output Other: The input port has no effect, only the ordinary input port

MODBUS address	Attributes	Defaults	Range	description
66	R/W	1	[0,11]	Output port 1 setting register
67	R/W	4	[0,11]	Output port 2 setting register
102	R/W	1	[0,11]	Output port 3 setting register (other products)
103	R/W	4	[0,11]	Output port 4 setting register (other products)

- When the output port 1/2 setting register value is set to 0 (normal output, user control function), the register with MODBUS address 68 is used to set whether the output port is turned on. It should be noted that the output port polarity in MODBUS address 66/67 still works. The MODBUS address 68 register is described as follows:

15			8
Reserve			
R-0			

7	2	1	0
Reserve		OUT1VAL	OUT0VAL
R-0		R/W-0	R/W-0

BIT	Name	Description
2～15	Reserve	Read always returns 0
1	OUT1VAL	Set the level status of the output port OUT1 0: Output port 1 is not conducting 1: Output port 1 is turned on
0	OUT0VAL	Set the level status of the output port OUT0 0: Output port 0 is not conducting 1: Output port 0 is turned on

MODBUS address	Attributes	Defaults	Range	description
68	R/W	0	[0,1]	Output state setting when OUT0 and OUT1 are used as normal outputs
69	R	-	-	Current input function valid flag (consistent with digital input port function) 0: The corresponding function is invalid. 1: Corresponding function is valid
104	R	-	-	Current output function valid flag (consistent with digital output port function) 0: The corresponding function is invalid. 1: Corresponding function is valid

2.6.10 Point motion parameter setting [70~74]

MODBUS address	Attributes	Defaults	Range	description
70	R/W	200	[10,1000]	Acceleration during point motion, unit: R/S^2
71	R/W	200	[10,1000]	Deceleration during point motion, unit: R/S^2
72	R/W	600	[0,3000]	Maximum speed during point motion, in RPM
73	R/W	2000	[-16777216,16777216]	Running pulse command during point motion, unit: number of pulses P73 is the lower 16 bits of data, P74 is the upper 16 bits of data
74				

The 73, 74 registers form a 32-bit signed register.

- In the incremental operation mode, the absolute values of 73 and 74 indicate the running distance. Write 1 or 2 through register 18 to control whether the motor is running forward or reverse.
- In the absolute position mode, the signed data consisting of 73 and 74 indicates the target position, and writing 1 by 18 causes the motor to run to the set distance.

2.6.11 Jog mode parameter setting [75~78]

MODBUS address	Attributes	Defaults	Range	description
75	R/W	100	[10,1000]	Jog acceleration, unit: R/S^2
76	R/W	100	[10,1000]	Jog deceleration, unit: R/S^2
77	R/W	600	[0,3000]	Jog speed, unit: RPM
78	R/W	500	[10,1000]	Emergency stop deceleration, unit: R/S^2

2.6.12 Internal pulse control parameters [84~89]

MODBUS address	Attributes	Defaults	Range	description
84	R/W	0	[0,1]	Internal pulse command operation mode 0: incremental position mode 1: absolute position mode
85	R/W	0	[0,1]	0: Write 0 is invalid, read returns 0 1: Internal pulse instruction counter is cleared
88	R/W	0	[0,1]	0: The over tolerance alarm is valid 1: Out of tolerance alarm is invalid
89	R/W	50	[0,500]	Servo mode-integral gain

2.6.13 Driver Basic Parameter Register [90~99]

MODBUS address	Attributes	Defaults	Range	description
90	R/W	0	[0,1]	0: Write 0 is invalid, read returns 0 1: Write 1 to save the current parameters
91	R/W	0	[0,1]	0: Write 0 is invalid, read returns 0 1: Write 1 will restore factory settings
92	-	-	-	The manufacturer retains the use and the user prohibits the writing of data.
93	R	-	-	Drive ID number
94	R	-	-	Drive version number
95	R	-	-	Non-label

2.6.14 Built-in speedometer parameter setting [100~120]

MODBUS address	Attributes	Defaults	Range	description
100	R/W	200	[0,65535]	IO switching effective time in speedometer and position table mode = setting value x 50us
101	R/W	1000	[0,3000]	Current step test current setting
105	R/W	0	[0,3000]	Internal speed 1, unit: RPM
106	R/W	100	[0,3000]	Internal speed 2, unit: RPM
107	R/W	200	[0,3000]	Internal speed 3, unit: RPM
108	R/W	300	[0,3000]	Internal speed 4, unit: RPM
109	R/W	400	[0,3000]	Internal speed 5, unit: RPM
110	R/W	500	[0,3000]	Internal speed 6, unit: RPM
111	R/W	600	[0,3000]	Internal speed 7, unit: RPM
112	R/W	700	[0,3000]	Internal speed 8, unit: RPM
113	R/W	800	[0,3000]	Internal speed 9, unit: RPM
114	R/W	900	[0,3000]	Internal speed 10, unit: RPM
115	R/W	1000	[0,3000]	Internal speed 11, unit: RPM
116	R/W	1100	[0,3000]	Internal speed 12, unit: RPM
117	R/W	1200	[0,3000]	Internal speed 13, unit: RPM
118	R/W	1300	[0,3000]	Internal speed 14, unit: RPM
119	R/W	1400	[0,3000]	Internal speed 15, unit: RPM
120	R/W	1500	[0,3000]	Internal speed 16, unit: RPM

2.6.15 Built-in position table parameter setting [121～156]

MODBUS address	Attributes	Defaults	Range	description
121	R	-	-	Currently triggered position table
122	R/W	100	[100,110]	Default parameter ID number (do not modify)
125 126	R/W	0	[-16777216,16777216]	Internal position 1 command P125 is the lower 16 bits, P126 is the upper 16 bits.
127 128	R/W	0	[-16777216,16777216]	Internal position 2 command P127 is the lower 16 bits, P128 is the upper 16 bits.
129 130	R/W	0	[-16777216,16777216]	Internal position 3 command P129 is the lower 16 bits, P130 is the upper 16 bits.
131 132	R/W	0	[-16777216,16777216]	Internal position 4 command P131 is the lower 16 bits, P132 is the upper 16 bits.
133 134	R/W	0	[-16777216,16777216]	Internal position 5 command P133 is the lower 16 bits, P134 is the upper 16 bits.
135 136	R/W	0	[-16777216,16777216]	Internal position 6 command P135 is the lower 16 bits, P136 is the upper 16 bits.
137 138	R/W	0	[-16777216,16777216]	Internal position 7 command P137 is the lower 16 bits, P138 is the upper 16 bits.
139 140	R/W	0	[-16777216,16777216]	Internal position 8 command P139 is the lower 16 bits, P140 is the upper 16 bits.
141 142	R/W	0	[-16777216,16777216]	Internal position 9 command P141 is the lower 16 bits, P142 is the upper 16 bits.

143 144	R/W	0	[-16777216,16777216]	Internal position 10 command P143 is the lower 16 bits, P144 is the upper 16 bits.
145 146	R/W	0	[-16777216,16777216]	Internal position 11 command P145 is the lower 16 bits, P146 is the upper 16 bits.
147 148	R/W	0	[-16777216,16777216]	Internal position 12 command P147 is the lower 16 bits, P148 is the upper 16 bits.
149 150	R/W	0	[-16777216,16777216]	Internal position 13 command P149 is the lower 16 bits, P150 is the upper 16 bits.
151 152	R/W	0	[-16777216,16777216]	Internal position 14 command P151 is the lower 16 bits, P152 is the upper 16 bits.
153 154	R/W	0	[-16777216,16777216]	Internal position 15 command P153 is the lower 16 bits, P154 is the upper 16 bits.
155 156	R/W	0	[-16777216,16777216]	Internal position 16 command P155 is the lower 16 bits, P156 is the upper 16 bits.

2.6.16 Torque mode register [157~158]

MODBUS address	Attributes	Defaults	Range	description
157	R/W	1000	[1,65535]	Torque mode speed loop proportional gain
158	R/W	15000	[0,65535]	Torque mode speed loop integral gain

2.6.17 Analog position control mode parameters [214~218]

MODBUS address	Attributes	Defaults	Range	description

214 215	R/W	4000	[0,0xFFFFF]	When the analog input voltage is set to 3.3V, the corresponding position command is set. 214 is the lower 16 bits of data, 215 is the upper 16 bits of data
216 217	R	-	-	Position command corresponding to the current input voltage 216 is the lower 16 bits of data, 217 is the upper 16 bits of data
218	R/W	5	[0,32767]	When the difference between the position command corresponding to the analog input voltage and the current position command is set within the set range, the position command is not adjusted. It is used to eliminate the jitter of the analog input voltage or the frequent jitter of the motor when the P214/215 parameter setting is relatively large.

2.6.18 Multi-segment position operation control mode parameters [221~271]

MODBUS address	Attributes	Defaults	Range	description

<table>
<tr><td>221</td><td>R/W</td><td>0</td><td>[0,2]</td><td>Set multi-segment position operation mode
0: Single operation mode
Start from the first stage displacement and run to the end point displacement number set by parameter P222, and then stop;
1: Cycle mode
Start from the first stage displacement to the end point number of the parameter set by parameter P222, and then start the cycle again from the first stage displacement;
2: Control mode by IN input signal
Select the displacement segment by the IN input function for "Multi-position position control 3/2/1/0"
<table>
<tr><th>Multi-segment control 3</th><th>Multi-segment control 2</th><th>Multi-segment control 1</th><th>Multi-segment control 0</th><th>displacement selection</th></tr>
<tr><td>OFF</td><td>OFF</td><td>OFF</td><td>OFF</td><td>第 1 段</td></tr>
<tr><td>OFF</td><td>OFF</td><td>OFF</td><td>ON</td><td>第 2 段</td></tr>
<tr><td>OFF</td><td>OFF</td><td>ON</td><td>OFF</td><td>第 3 段</td></tr>
<tr><td>……</td><td>……</td><td>……</td><td>……</td><td>……</td></tr>
<tr><td>ON</td><td>ON</td><td>ON</td><td>ON</td><td>第 16 段</td></tr>
</table>
</td></tr>
<tr><td>222</td><td>R/W</td><td>16</td><td>[1,16]</td><td>Set the number of multi-segment displacement end points,
● The parameter takes effect only when the parameter P221 is set to 0/1.</td></tr>
<tr><td>223</td><td>R/W</td><td>0</td><td>[0,1]</td><td>Set the unit of waiting time after the end of each displacement operation
0:ms
1:s
The parameter takes effect only when the parameter P221 is set to 0/1.</td></tr>
<tr><td>224</td><td>R/W</td><td>100</td><td>[0,3000]</td><td>1st stage displacement maximum running speed, unit RPM
● For the displacement stroke, please refer to the</td></tr>
</table>

				built-in position table parameter setting [121～156] “Internal position 1” setting.
225	R/W	100	[1,2000]	The first stage displacement acceleration and deceleration, unit: R/S^2
226	R/W	100	[0,65535]	Waiting time after the end of the first stage displacement ● The parameter takes effect only when the parameter P221 is set to 0/1.
227	R/W	100	[0,3000]	2nd stage displacement maximum running speed, unit RPM
228	R/W	100	[1,2000]	The second stage displacement acceleration and deceleration, unit: R/S^2
229	R/W	100	[0,65535]	Waiting time after the end of the second stage displacement
230	R/W	100	[0,3000]	The third stage displacement maximum running speed, unit RPM
231	R/W	100	[1,2000]	Stage 3 displacement acceleration, deceleration, unit: R/S^2
232	R/W	100	[0,65535]	Waiting time after the end of the third stage displacement
233	R/W	100	[0,3000]	Stage 4 displacement maximum running speed, unit RPM
234	R/W	100	[1,2000]	Stage 4 displacement acceleration, deceleration, unit: R/S^2
235	R/W	100	[0,65535]	Waiting time after the end of the fourth stage displacement
236	R/W	100	[0,3000]	5th stage displacement maximum running speed, unit RPM
237	R/W	100	[1,2000]	The fifth stage displacement acceleration and deceleration, unit: R/S^2

238	R/W	100	[0,65535]	Waiting time after the end of the fifth stage displacement
239	R/W	100	[0,3000]	Stage 6 displacement maximum running speed, unit RPM
240	R/W	100	[1,2000]	Stage 6 displacement acceleration, deceleration, unit: R/S^2
241	R/W	100	[0,65535]	Waiting time after the end of the sixth stage displacement
242	R/W	100	[0,3000]	7th stage displacement maximum running speed, unit RPM
243	R/W	100	[1,2000]	7th displacement acceleration and deceleration, unit: R/S^2
244	R/W	100	[0,65535]	Waiting time after the end of the 7th displacement
245	R/W	100	[0,3000]	Stage 8 displacement maximum running speed, unit RPM
246	R/W	100	[1,2000]	The eighth stage displacement acceleration and deceleration, unit: R/S^2
247	R/W	100	[0,65535]	Waiting time after the end of the eighth stage displacement
248	R/W	100	[0,3000]	Stage 9 displacement maximum running speed, unit RPM
249	R/W	100	[1,2000]	9th displacement acceleration and deceleration, unit: R/S^2
250	R/W	100	[0,65535]	Waiting time after the end of the 9th displacement
251	R/W	100	[0,3000]	Stage 10 displacement maximum running speed, unit RPM
252	R/W	100	[1,2000]	Stage 10 displacement acceleration, deceleration, unit: R/S^2
253	R/W	100	[0,65535]	Waiting time after the end of the 10th displacement

254	R/W	100	[0,3000]	Stage 11 displacement maximum running speed, unit RPM
255	R/W	100	[1,2000]	11th displacement acceleration and deceleration, unit: R/S^2
256	R/W	100	[0,65535]	Waiting time after the end of the 11th displacement
257	R/W	100	[0,3000]	Stage 12 displacement maximum running speed, unit RPM
258	R/W	100	[1,2000]	Stage 12 displacement acceleration, deceleration, unit: R/S^2
259	R/W	100	[0,65535]	Waiting time after the end of the 12th displacement
260	R/W	100	[0,3000]	Stage 13 displacement maximum running speed, unit RPM
261	R/W	100	[1,2000]	Stage 13 displacement acceleration, deceleration, unit: R/S^2
262	R/W	100	[0,65535]	Waiting time after the end of the 13th displacement
263	R/W	100	[0,3000]	Stage 14 displacement maximum running speed, unit RPM
264	R/W	100	[1,2000]	Stage 14 displacement acceleration, deceleration, unit: R/S^2
265	R/W	100	[0,65535]	Waiting time after the end of the 14th displacement
266	R/W	100	[0,3000]	Stage 15 displacement maximum running speed, unit RPM
267	R/W	100	[1,2000]	The 15th paragraph displacement acceleration, deceleration, unit: R / S ^ 2
268	R/W	100	[0,65535]	Waiting time after the end of the 15th displacement
269	R/W	100	[0,3000]	Stage 16 displacement maximum running speed, unit RPM
270	R/W	100	[1,2000]	61st displacement acceleration and deceleration, unit: R/S^2

271	R/W	100	[0,65535]	Waiting time after the end of the 16th displacement

2.6.19 Analog input parameter setting [272~279]

MODBUS address	Attributes	Defaults	Range	description
272	R/W	0	[0,1650]	Set the analog input voltage offset in mV
273	R/W	10	[0,2000]	Set the analog input voltage low pass filter cutoff frequency, unit: Hz
274	R/W	50	[0,1000]	Set the analog input voltage dead band, unit: mV
275	R/W	0	[0,1000]	Set the analog input voltage zero drift, unit: mV
276	R/W	100	[0,3000]	Set the analog input voltage to 3.3V, the corresponding speed, unit: RPM
277	R	-	-	DSP current sampling voltage value, unit: mV
278	R	-	-	Analog input voltage value after zero drift, dead zone, and offset processing, unit: mV
279	R	-	-	Current speed corresponding to the analog input voltage, in RPM

2.6.20 Modbus communication error counter [280~282]

MODBUS address	Attributes	Defaults	Range	description
280	R/W	-	-	Modbus bus error counter Read: Number of Modbus bus errors since the last time the counter was reset Write: Reset Modbus bus error counter
281	R/W	-	-	Modbus CRC error counter Read: Number of Modbus CRC errors since the last

				time the counter was reset Write: Reset Modbus CRC Error Counter
282	R/W	-	-	Modbus Receive Byte Count Error Counter Read: Modbus Receive Byte Error Counter from Last Reset Counter Write: Reset Modbus Receive Bytes Error Counter

2.6.21 Homing Control Mode Setting [287~298]

<table>
<tr><th>MODBUS address</th><th>Attributes</th><th>Defaults</th><th>Range</th><th>description</th></tr>
<tr><td>287</td><td>R/W</td><td>1</td><td>[0,6]</td><td>Set the home position return enable control mode
<table>
<tr><th>Set value</th><th>Control method</th></tr>
<tr><td>0</td><td>Prohibit the origin return function</td></tr>
<tr><td>1</td><td>Use the IN input function to trigger the mechanical homing function for the IN terminal that starts the zero return</td></tr>
<tr><td>2</td><td>Use the IN input function to trigger the electrical homing function for the IN terminal that initiates zero return
The electrical homing point is generally used after the mechanical homing point and does not require a sensor input signal. Directly return to the position command set by parameter P293/294 according to the absolute position. After the electrical return to the origin, the parameter P8/9 is equal to the parameter P293/294.</td></tr>
<tr><td>3</td><td>Power-on automatic mechanical homing
Set to this value, and by writing 1 to P90 parameter to save the parameters permanently, the next time you power on, it will automatically return to the origin.
Trigger homing only after power-on and when the motor is enabled</td></tr>
<tr><td>4</td><td>Communication triggers mechanical homing function
When the motor is enabled, writing this value</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td></td><td>
<table>
<tr><td></td><td>will immediately trigger the mechanical homing function. This register is cleared after the homing is completed.</td></tr>
<tr><td>5</td><td>Communication triggers electrical homing function
When the motor is enabled, writing this value will immediately trigger the electrical homing function. This register is cleared after the homing is completed.</td></tr>
<tr><td>6</td><td>Communication triggers the current position as the origin
When the motor is enabled, the value is written and the drive will use the current position as the origin. This register is cleared after the homing is completed</td></tr>
</table>
</td></tr>
<tr><td>288</td><td>R/W</td><td>0</td><td>[0,5]</td><td>
Set the origin return mode
<table>
<tr><th>Set value</th><th>Control method</th></tr>
<tr><td>0</td><td>Positive homing
Deceleration point: origin switch
Origin: origin switch</td></tr>
<tr><td>1</td><td>Negative homing
Deceleration point: origin switch
Origin: origin switch</td></tr>
<tr><td>2</td><td>Positive homing
Deceleration point: positive limit switch
Origin: forward limit switch</td></tr>
<tr><td>3</td><td>Negative homing
Deceleration point: negative limit switch
Origin: negative limit switch</td></tr>
<tr><td>4</td><td>Positive homing
Deceleration point: mechanical limit position
Origin: mechanical limit position</td></tr>
<tr><td>5</td><td>Positive homing
Deceleration point: mechanical limit position
Origin: mechanical limit position</td></tr>
</table>
</td></tr>
<tr><td>289</td><td>R/W</td><td>50</td><td>[0,1000]</td><td>High-speed search for the speed of the origin switch</td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td></td><td>signal, unit: RPM</td></tr>
<tr><td>290</td><td>R/W</td><td>10</td><td>[0,1000]</td><td>Searching for the speed of the home switch signal at low speed, in RPM</td></tr>
<tr><td>291</td><td>R/W</td><td>200</td><td>[1,1000]</td><td>Search for the acceleration and deceleration of the origin switch signal, unit: R/S^2</td></tr>
<tr><td>292</td><td>-</td><td>-</td><td>-</td><td>Reserved</td></tr>
<tr><td>293</td><td rowspan="2">R/W</td><td rowspan="2">0</td><td rowspan="2">[-1048576,1048576]</td><td rowspan="2">Set the machine origin offset, unit: command pulse
Note: When the parameter setting of P293/294 is positive, it means forward running.</td></tr>
<tr><td>294</td></tr>
<tr><td>295</td><td>R/W</td><td>0</td><td>[0,1]</td><td>Mechanical origin offset and limit processing：
<table>
<tr><th>Set value</th><th>Machine origin offset and limit processing</th></tr>
<tr><td>0</td><td>P293/P294 is the coordinate after the return of origin.
Recover the origin after re-triggering the origin return enable
Remarks:
Mechanical origin: the mechanical origin does not coincide with the mechanical zero point. After the origin switch signal is found, the current position P8/9 parameter is forced to the P293/294 parameter setting value.
Limit processing method: the origin return trigger signal is given again, and the motor direction performs the origin return</td></tr>
<tr><td>1</td><td>P293/P294 is the relative offset after the return of origin.
Recover the origin after re-triggering the origin return enable
Remarks:
Mechanical origin: The mechanical origin coincides with the mechanical zero point. After finding the origin switch signal, the motor stops after the command stroke set by parameter P293/394, and the parameter P8/9 is equal to the parameter setting value of P293/P294.
Limit processing method: the origin return trigger signal is given again, and the motor</td></tr>
</table></td></tr>
</table>

<table>
<tr><td></td><td></td><td></td><td></td><td>
<table>
<tr><td></td><td>direction performs the origin return</td></tr>
<tr><td>2</td><td>P293/P294 is the coordinate after the return of origin.
Encounter limit automatically reverse to find the origin
Remarks:
Mechanical origin: the mechanical origin does not coincide with the mechanical zero point. After the origin switch signal is found, the current position P8/9 parameter is forced to the P293/294 parameter setting value.
Limit processing method: automatic reverse execution to resume homing</td></tr>
<tr><td>3</td><td>P293/P294 is the relative offset after the return of origin.
Encounter limit automatically reverse to find the origin
Remarks:
Mechanical origin: The mechanical origin coincides with the mechanical zero point. After finding the origin switch signal, the motor stops after the command stroke set by parameter P293/394, and the parameter P8/9 is equal to the parameter setting value of P293/P294.
Limit processing method: automatic reverse execution to resume homing</td></tr>
</table>
</td></tr>
<tr><td>296</td><td>R/W</td><td>5000</td><td>[1000,65535]</td><td>When P288 is set to 4/5, the collision is enabled to return to the origin. When the running speed of the motor is lower than the parameter setting value of P297, and the actual motor current is greater than or equal to the parameter setting value of P298, it is considered that the mechanical limit position is reached. At this time, the internal collision return to the origin counter starts counting, and the counter time is greater than P296. When the set value is set, the motor returns to the home position.
Set the collision return origin detection time, unit: 50us</td></tr>
<tr><td>297</td><td>R/W</td><td>5</td><td>[1,1000]</td><td>Set the collision back to the origin detection speed, unit:</td></tr>
</table>

				PRM
298	R/W	1000	[1,6000]	Set the magnitude of the collision back to the origin torque, unit: mA

Chapter 3 Modbus/RTU Routines

3.1 Origin reset related settings

3.1.1 Introduction to Functions

Origin: The mechanical origin can be expressed as the origin switch signal or limit switch signal, which is set by parameter P288.

Zero point: the positioning target point, which can be expressed as the origin + offset (P293/P294 setting). When the offset is set to 0, the zero coincides with the origin.

The home position return function is to activate the zero point and complete the positioning function after the home position return function is triggered in the drive enable state.

During the return-to-origin operation, other position commands (including the re-triggered home position return enable signal) are masked; after the home return operation is completed, the drive can respond to other position commands.

The zero return function includes two modes: zero return and zero return.

Zero return of the origin: After receiving the return-to-origin trigger signal, the drive actively locates the relative position of the motor shaft and the machine origin according to the preset mechanical origin. First, the origin is searched, and then the offset is moved to the zero position based on the origin. The zero return of the origin is usually used for the first time to find the zero point.

Electrical zero return: After the zero position is determined by the zero return operation of the origin, the relative displacement is moved with the current position as the starting point.

After the return-to-origin is completed (including zero return and electrical zero return), the current position of the motor (P8/P9) is consistent with the mechanical origin offset (P293/P294).

After the return of the origin is completed, the driver outputs the zero return completion signal of the origin, and the host computer can confirm the completion of the origin return after receiving the signal. For the function setting of the output port, please refer to the output port setting register [66~69].

3.1.2 Zero return to origin

Take the following situation as an example to illustrate the zero return of the origin:

- Positive zero return, deceleration point, origin is the origin switch (P288=0)
- Positive zero return, deceleration point and origin are positive limit switches (P288=2)
- Positive zero return, deceleration point and origin are mechanical limit positions (P288=4)

(1) Zero return of origin: positive return to zero, deceleration point and origin are origin switch (P288=0)

① The origin switch (deceleration point) signal is invalid when the motor starts to move (0-invalid, 1-valid), the positive limit switch is not triggered in the whole process.

The motor first searches for the deceleration point signal at the high speed forward set by P289 until the rising edge of the deceleration point is encountered. The deceleration set according to P291 deceleration to 0, and then the reverse acceleration to the low speed search deceleration point set by -P290. When the falling edge of the signal encounters the falling edge of the deceleration point signal, it will stop immediately. Then, P290 will continue to search for the rising edge of the origin signal at low speed. In the forward acceleration or forward constant speed operation, it will stop immediately when it encounters the rising edge of the origin signal.

减速点/原点：**Deceleration point/origin**

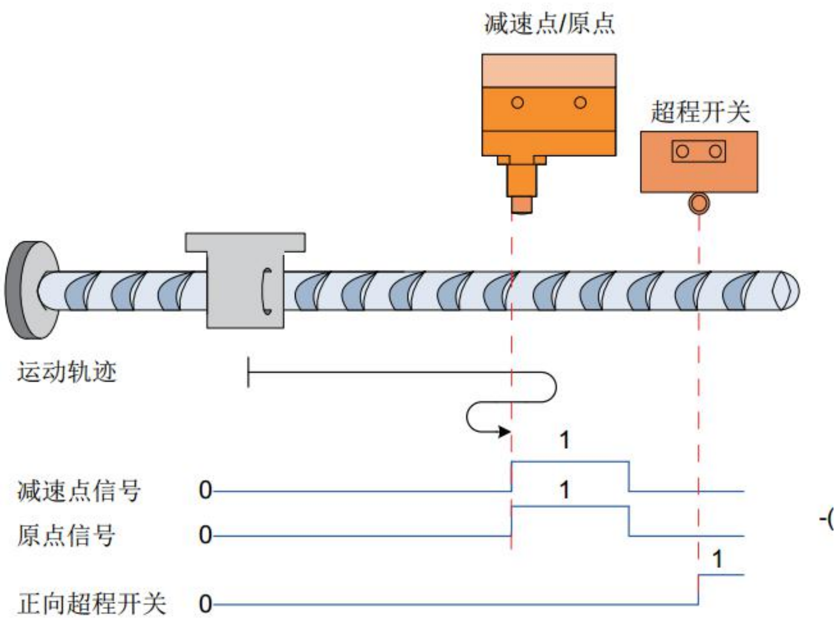
超程开关：**Overshoot switch**

运动轨迹： **path of particle**

减速点信号： **Deceleration point signal**

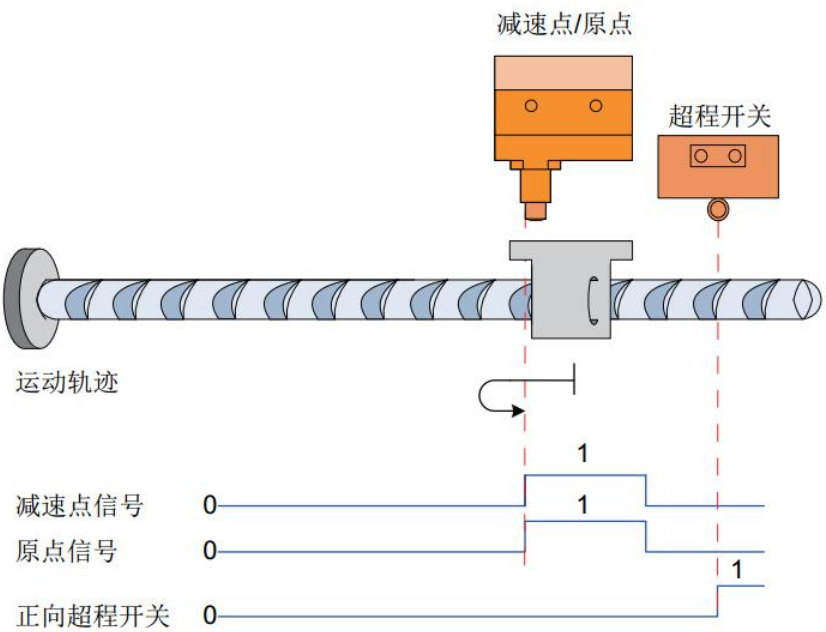
原点信号： **Origin signal**

正向超程开关：**Forward Overrun Switch**



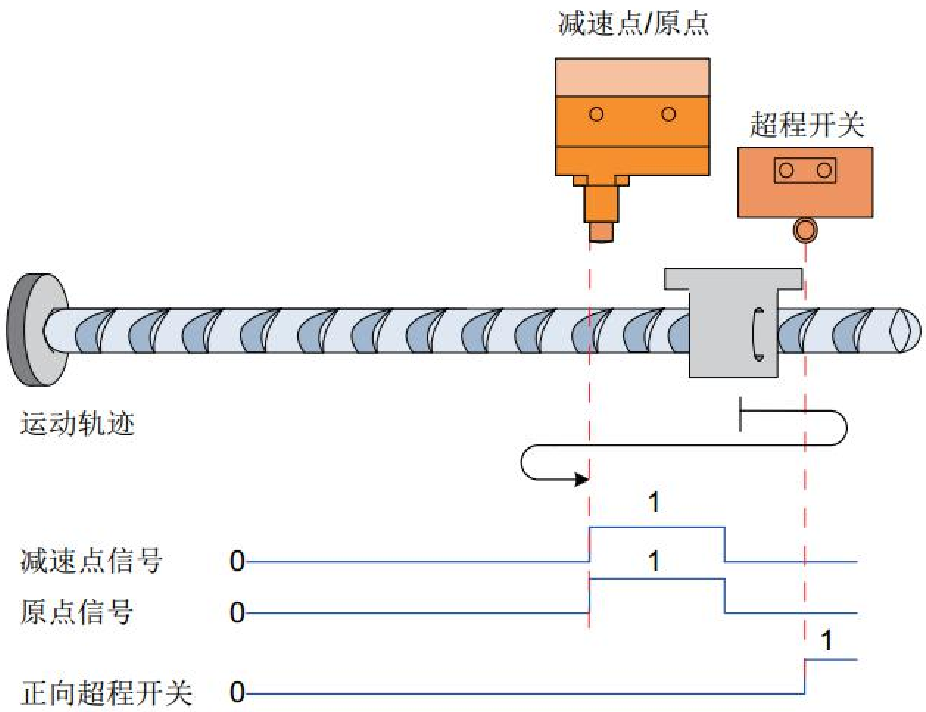
② When the motor starts running, the origin switch (deceleration point) signal is valid, and the positive limit switch is not triggered during the whole process.

The motor directly searches for the falling edge of the deceleration point signal with the -P290 set value low speed. When the falling edge of the deceleration point signal is encountered, it will stop immediately. Then, the P290 set value will continue to search for the rising edge of the origin signal, positive acceleration or forward direction. In constant speed operation, it immediately stops when it encounters the rising edge of the origin signal.



③ When the motor starts running, the origin switch (deceleration point) signal is invalid, and the positive limit switch is activated during the process.

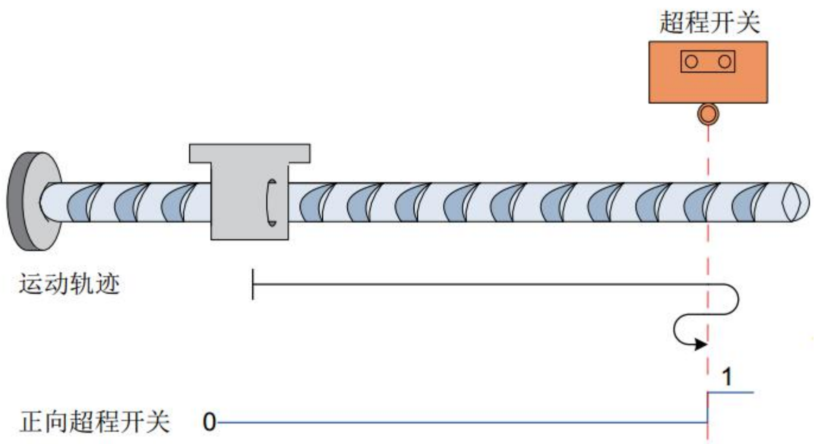
The motor firstly searches for the deceleration point signal at the high speed of P289 setting value. After the positive limit switch is encountered, the driver decides to immediately return to zero according to the setting of P295 (P295=2 or 3), or stop and wait for the upper computer to restart again. The origin return zero trigger signal (P295=0 or 1) is given. After the condition is satisfied, the driver searches for the falling edge of the deceleration point signal with -P289 reverse high speed. After the falling edge of the deceleration point signal is encountered, the deceleration is decelerated according to the deceleration set by P291. After 0, the rising edge of the origin signal is searched for in the forward low speed according to the set value of P290. During the forward acceleration or forward constant running, the rising edge of the origin signal is stopped immediately.



(2) Zero return of origin: positive return to zero, deceleration point, origin is forward limit switch (P288=2)

① The positive limit switch signal is invalid when the motor starts moving (0-invalid, 1-valid)

The motor first searches for the positive limit switch at the high speed with the P289 set value. After the rising edge of the positive limit switch signal, it decelerates to 0 according to the deceleration set by P291, and then reverses with the -P290 set value. To search for the falling edge of the positive limit switch signal, stop immediately after encountering the falling edge of the positive limit switch signal, then resume the forward running, and search for the rising edge of the positive limit switch signal with the P290 set value forward low speed. During forward acceleration or forward constant speed operation, it immediately stops when it encounters the rising edge of the positive limit switch signal.



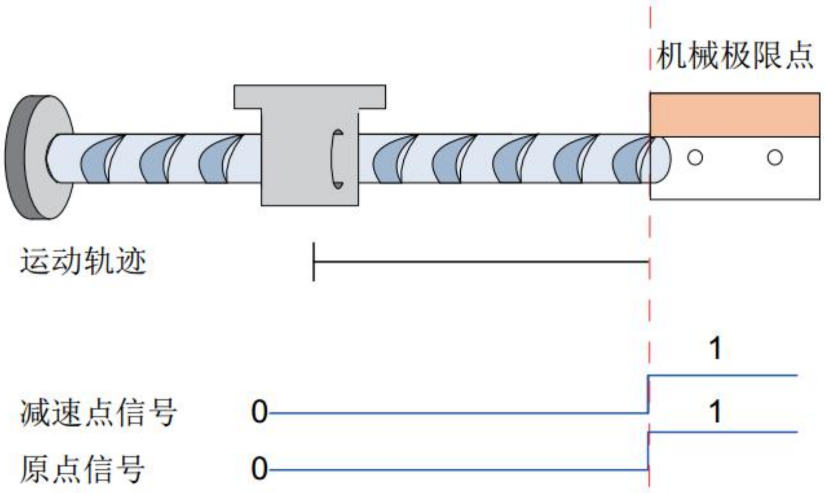
② The positive limit switch signal is valid when the motor starts moving.

The motor directly searches for the falling edge of the positive limit switch signal with the -P290 set value reverse speed. When the falling edge of the positive limit switch signal is encountered, it stops immediately, and then searches for the positive limit with the P290 set value low speed forward. On the rising edge of the switching signal, during the forward acceleration or the forward constant speed operation, the rising edge of the positive limit switch signal is stopped immediately.

(3) Zero return to origin: positive zero return, deceleration point, origin is positive limit switch (P288=4)

The motor first runs at a low speed with the P290 set value. After hitting the mechanical limit position, if the motor torque reaches the P298 torque upper limit and the motor speed is lower than the P297 set value, this state is maintained after the P296 set value time. In order to reach the mechanical limit position, the motor immediately stops.

Note: This zero return mode (P288=4/5) is only available in closed loop mode.



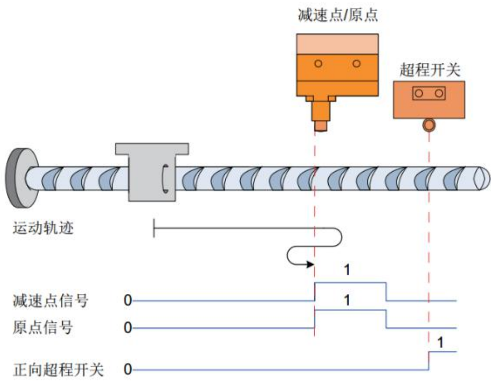
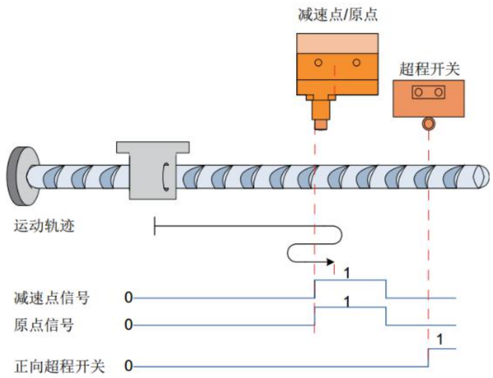
3.1.3 Electrical return to zero

After the zero return of the origin is completed, the mechanical zero position of the system is known. At this time, after setting P293/P294, the motor can be moved from the current position (P8/P9) to the specified position (P293/P294). In the electrical zero return mode, the whole motor runs at high speed with the P289 set value. The total motor displacement is determined by the difference between P293/P294 and P8/P9. The running direction is determined by the positive and negative of the total motor displacement. The displacement command is completed and the motor is immediately Stop.

3.1.4 Mechanical origin and mechanical zero

Take P288=0 as an example to illustrate the difference between mechanical origin and mechanical zero.

The mechanical origin does not coincide with the mechanical zero	Mechanical origin coincides with mechanical zero
If the origin offset (P293/P294≠0) is set and the machine origin does not coincide with the mechanical zero point (P295=0), during	If the origin offset (P293/P294≠0) is set and the machine origin coincides with the mechanical zero point (P295=1), during

the positive acceleration or forward constant speed operation, the rising edge of the origin signal is stopped immediately, and the machine stops. The rear motor current position P8/P9 is forced to the P293/P294 setting.	forward acceleration or forward constant speed operation, the motor continues to move after the rising edge of the origin signal is reached. The current absolute position P8/P9 is the set value of P293/P294.
	

For the specific parameter setting and address of zero return, please refer to the homing control mode setting [287~298]

3.2 Communication Control Mode

In this mode, the user can run the specified pulse stroke or jog operation by communicating the given run command. The details are as follows.

3.2.1 Point control mode

The NT86/NR60 has the function of communication control motor running specified pulse stroke. The specific modes and parameters to be set are as follows (register addresses are not specified or specified as decimal numbers):

(1) Set the value of register address 20 (preset application selection in internal pulse mode) to 0 (communication control, response register address 18 command);

(2) Set the function of the digital input and output port according to the application requirements and the actual terminal block;

(3) Set the motion parameters:

address	Unit	Parameter Description
70	R/S^2	Acceleration of point motion
71	R/S^2	Deceleration of point motion
72	RPM	Speed of point movement
73	Command pulse	The number of instruction pulses for point motion is lower than the 16-bit register
74	Command pulse	The number of instruction pulses for point motion is high 16-bit register
78	R/S^2	Emergency stop deceleration
84	-	Set the location mode: 0: incremental 1: Absolute

(4) Communication given run command: Start the bit movement by writing the value 1 (fixed length forward) and 2 (fixed length reverse) to the register 18. (For details on this register, please see "Drive Control Mode" Set register 18 in [17~23]";

(5) If you need to stop during operation, you can write a value of 6 to the register 18 (deceleration stop, deceleration is the set value of register 71), value 5 (emergency stop stop, deceleration is the set value of register 78) .

- Note:
- The motor is in operation and only responds to the stop command (deceleration stop or emergency stop). If you need to change the motor running direction by command, you need to send a stop command to wait for the motor to stop before sending another direction start signal.
- The acceleration (register 70), deceleration (register 71), and speed (register 72) are changed during motor operation, but the drive does not respond to these settings immediately. It needs to be set after the motor is stopped again. Set to run. It is important to note that the emergency stop deceleration (Register 78) is responded to during the current sport emergency stop, without waiting for the next stop of the emergency stop.

3.2.2 Jog control mode

The NT86/NR60 has the function of controlling the jog operation of the motor through communication. The specific modes and parameters to be set are as follows (register addresses are not specified or specified as decimal numbers):

(1) Set the value of register address 20 (preset application selection in internal pulse mode) to 0 (communication control, response register address 18 command);

(2) Set the function of the digital input and output port according to the application requirements and the actual terminal block;

(3) Set the motion parameters:

address	Unit	Parameter Description
75	R/S^2	Acceleration of jog motion
76	R/S^2	Jog speed deceleration
77	RPM	Jog speed
78	R/S^2	Emergency stop deceleration

(4) Communication given run command: Start the bit movement by writing the value 3 (continuous forward rotation) and 4 (continuous reverse rotation) to the register 18. (For details on this register, please see "Drive Control Mode Setting [Register 18 in 17~23]";

(5) If you need to stop during operation, you can write a value of 6 to the register 18 (deceleration stop, deceleration is the set value of the register 76), value 5 (emergency stop, deceleration is the set value of the register 78) .

- Note:
- The motor is in operation and only responds to the stop command (deceleration stop or emergency stop). If you need to change the motor running direction by command, you need to send a stop command to wait for the motor to stop before sending another direction start signal.
- The acceleration (register 75) and deceleration (register 76) are changed while the motor is running, but the drive does not respond to these settings immediately. It needs to be started after the motor stops and will run at the set

value. It is important to note that the emergency stop deceleration (Register 78) is responded to during the current sport emergency stop, without waiting for the next stop of the emergency stop.

The speed can be changed during the running of the motor (Register 77), and the drive will respond immediately, ie the motor will immediately run at the set speed value without responding after restarting after stopping.

3.3 IO Control: Start and Stop + Direction

The NT86/NR60 can use this mode to control the operation of the motor using two IN ports. One of the IN terminals is used to control the start/stop of the motor, and one IN terminal is used to control the running direction of the motor. The specific settings are as follows:

(1) Set the value of register address 20 (preset application selection in internal pulse mode) to 2 (start and stop + direction mode);

(2) Set the function of the digital input and output port according to the application needs and the actual terminal block. Among them, please set the function of the two IN terminals to “jog forward/start/stop” and “jog reverse/direction” to control the start/stop and running direction of the motor. For the IN terminal function setting, please refer to "Input Port Setting Register [60~65]";

(3) Set the motion parameters:

address	Unit	Parameter Description
75	R/S^2	Acceleration of jog motion
76	R/S^2	Jog speed deceleration
77	RPM	Jog speed
78	R/S^2	Emergency stop deceleration

(4) Input the appropriate level through the corresponding IN port to control the operation and direction of the motor.

- Note:
- Acceleration (Register 75), Deceleration (Register 76), Speed (Register 77),

Emergency Stop (Register 78) can be dynamically changed while the motor is running, and the drive will respond to these settings immediately.

- The direction signal can be switched during the running of the motor. At this time, the motor will decelerate to stop at the deceleration set by register 75 and then accelerate to the set speed in the opposite direction.

3.4 IO Control: Forward + Reverse

The NT86/NR60 can use this mode to control the operation of the motor using two IN ports. One of the IN terminals is used to control the forward rotation of the motor, and one IN terminal is used to control the reverse rotation of the motor. The specific settings are as follows:

(1) Set the value of register address 20 (preset application selection in internal pulse mode) to 3 (forward + reverse mode);

(2) Set the function of the digital input and output port according to the application needs and the actual terminal block. Among them, please set the function of the two IN terminals to “jog forward/start/stop” and “jog reverse/direction” to control the forward and reverse movement of the motor. For the IN terminal function setting, please refer to "Input Port Setting Register [60~65]";

(3) Set the motion parameters:

address	Unit	Parameter Description
75	R/S^2	Acceleration of jog motion
76	R/S^2	Jog speed deceleration
77	RPM	Jog speed
78	R/S^2	Emergency stop deceleration

(4) Input the appropriate level through the corresponding IN port to control the forward and reverse movement of the motor

- Note:
- The user can dynamically change the acceleration (Register 75), deceleration (Register 76), speed (Register 77), emergency stop (Register 78) while the motor is

running, and the drive will respond to these settings immediately.

- Change the running direction during motor operation. Please cancel the running signal in this direction and wait for the motor to stop before giving the running signal in the other direction.

3.5 IO Control Speedometer Mode

This mode selects 16 speeds with up to 4 IOs. Normally, the first speed is set to 0, indicating that the motor is stopped.

After switching the IO state, the new speed takes effect after the time set by register 100.

The related registers are as follows:

parameter	Unit	RTU register address	Routine setting
Jog acceleration	R/S^2	40076（0x004B）	100（0x0064）
Jog deceleration	R/S^2	40077（0x004C）	100（0x0064）
Deceleration during emergency stop	R/S^2	40079（0x004E）	500（0x01F4）
IN1 port function	-	40077（0x003C）	46（0x002E）
IN2 port function	-	40077（0x003D）	47（0x002E）
IN3 port function	-	40077（0x003E）	48（0x002E）
IN4 port function	-	40077（0x003F）	49（0x002E）
Effective time after IO switch	50us	40101（0x0064）	200(时间 =200*50us=1ms)
Speed table 0	RPM	40106（0x0069）	0
Speed table 1	RPM	40107（0x0070）	100
Speed table 2	RPM	40108（0x0070）	200
Speed table 3	RPM	40109（0x0072）	300
Speed table 4	RPM	40110（0x0073）	400
Speed table 5	RPM	40111（0x0074）	500
Speed table 6	RPM	40112（0x0075）	600

Speed Table 7	RPM	40113（0x0076）	700
Speed Table 8	RPM	40114（0x0077）	800
Speed Table 9	RPM	40115（0x0078）	900
Speed table 10	RPM	40116（0x0079）	1000
Speed table 11	RPM	40117（0x007A）	1100
Speed table 12	RPM	40118（0x007B）	1200
Speed table 13	RPM	40119（0x007C）	1300
Speed table 14	RPM	40120（0x007D）	1400
Speed table 15	RPM	40121（0x007E）	1500

Step 1: No. 20 register setting APP control mode: 4

Step2: Set the acceleration and deceleration.

Write message: 01 10 00 69 00 10 20 00 00 00 64 00 C8 01 2C 01 90 01 F4 02 58 02 BC 03 20 03 84 03 E8 04 4C 04 B0 05 14 05 78 05 DC 03 92

Feedback message: 01 10 00 69 00 10 11 D9

Step3: Set the IO port and polarity for selecting the speed table. The functions of the IN1, IN2, IN3, and IN4 ports should be set to: internal speed control 0, 1, 2, 3, corresponding register value bits 46, 47, 48, 49.

Write message: 01 10 00 3C 00 04 08 00 2E 00 2F 00 30 00 31 3C 35

Feedback message: 01 10 00 3C 00 04 01 C6

Step4: Input the appropriate level at the corresponding IO port to control the motor operation.

The user can dynamically modify the speed table and acceleration and deceleration information during the running process.

The user can also use an input port to control the direction in which the motor is running. The function of this port should be set to: the internal speed command is reversed.

When the user's motor is running, the direction signal is switched, and the motor will first decelerate to stop and then accelerate to the set speed in the opposite direction.

3.6 IO Control Location Table Mode

Set the same way as 7.5

3.7 Internal Pulse Application Mode 20

The internal pulse application mode 20 integrates multiple application modes, in which IN jog, IN point, communication jog, communication point, multi-segment position operation, etc. can be realized. The specific settings are as follows:

3.7.1 Realizing the related settings of jog forward and reverse

(1) Set the acceleration, deceleration, speed, and emergency stop deceleration of the jog: set the corresponding value as described in Jog mode parameter setting [75~78];

(2) Set the corresponding IN pin function: Input port setting register [60 ~ 65]

IN pin	Settings	
	Polarity	Function bit
INx	0/1 (set according to input polarity)	7 (jog forward/start/stop)
INx	0/1 (set according to input polarity)	8 (jog inversion / direction)

(3) Startup method

The motor can be rotated forward/reverse by the PLC or the button to give the corresponding IN pin a level trigger signal.

By 485 communication, write 3 (jog forward), 4 (jog reverse), 5 (emergency stop), and 6 (deceleration stop) to the P18 register to realize the motor forward/reverse rotation. ;

Through the 485 communication, flip the “Polarity” bit in the corresponding IN pin configuration register to simulate an external IN trigger signal to achieve the motor's jog forward/reverse;

(4) During jog operation, the driver can respond to parameters such as acceleration, deceleration and speed modified by 485 communication in real time.

3.7.2 Implementation of the relevant settings for point inversion

(1) Set the acceleration, deceleration, speed, and stroke of the point: Please set the corresponding value as described in the point motion parameter setting [70~74];

(2) Set the emergency stop deceleration during point movement: Please set the corresponding value as described in Jog mode parameter setting [75~78];

(3) Set the position command in the movement of the point position. Operation mode P84 No. Parameter: Please refer to the description in the internal pulse control parameters [84~89] to set the corresponding value.

(4) Set the corresponding IN pin function: Input port setting register [60 ~ 65]

IN pin	Setting	
	Polarity	Function bit
INx	0/1 (set according to input polarity)	22 (USER1: forward)
INx	0/1 (set according to input polarity)	23 (USER2: reverse)

(5) Startup method

The edge of the motor can be forward/reverse by the PLC or the button to give the corresponding IN pin an edge trigger signal.

By 485 communication, write 1 (point forward), 2 (point reverse), 5 (emergency stop), and 6 (deceleration stop) to the P18 register to realize the positive/reverse position of the motor. ;

Through the 485 communication, flip the "Polarity" bit in the corresponding IN pin configuration register to simulate an external IN trigger signal to realize the positive/reverse position of the motor.

3.7.3 Jog Start and Stop + Direction Control Mode Related Settings

(1) Set the acceleration, deceleration, speed, and emergency stop deceleration of

the jog: set the corresponding value as described in Jog mode parameter setting [75~78];

(2) Set the corresponding IN pin function: Input port setting register [60 ~ 65];

IN pin	Setting	
	Polarity	Function bit
Inx	0/1 (set according to input polarity)	25 (USER4: start and stop)
Inx	0/1 (set according to input polarity)	14 (multi-speed control 0: direction)

(3) Startup method

The jog start/stop + direction control mode of the motor can be realized by a PLC or a button to give a corresponding level trigger signal to the corresponding IN pin.

Through the 485 communication, flip the "Polarity" bit in the corresponding IN pin configuration register to simulate an external IN trigger signal to realize the jog start/stop + direction control mode of the motor;

(4) During jog operation, the driver can respond to parameters such as acceleration, deceleration and speed modified by 485 communication in real time.

3.7.4 Implementing related settings for multi-segment position control mode

(1) Set the operation mode of the position table, the number of running end points, and the time unit: Please refer to the register description in the multi-segment position operation control mode parameter [221~271] to set the corresponding value;

(2) Set the stroke, acceleration and deceleration, speed, waiting time, etc. of each position: built-in position table parameter setting [121 ~ 156], multi-segment position operation control mode parameters [221 ~ 271];

(3) Set the corresponding IN pin function: Input port setting register [60 ~ 65]

When parameter P221 is set to 0/1: single sequential operation stop/cycle sequential operation mode

IN pin	Setting	
	Polarity	Function bit
Inx	0/1 (set according to input polarity)	24 (USER3: multi-segment position

		start signal)

In this mode of operation, the trigger signal is a level signal.

- When parameter P221 is set to 2: INx controls the switching mode of multi-segment position

IN pin	Settings	
	Polarity	Function bit
INx	0/1 (set according to input polarity)	24 (USER3: multi-segment position start signal)
INx	0/1 (set according to input polarity)	18 (multi-segment position control 0)
INx	0/1 (set according to input polarity)	19 (multi-segment position control 1)
INx	0/1 (set according to input polarity)	20 (multi-segment position control 2)
INx	0/1 (set according to input polarity)	21 (multi-segment position control 3)

The relationship between the INx function and the selected multi-segment position is as follows:

Multi-segment position control 3	Multi-segment position control 2	Multi-segment position control 1	Multi-segment position control 0	Multiple positions
OFF	OFF	OFF	OFF	1
OFF	OFF	OFF	ON	2
OFF	OFF	ON	OFF	3
……	……	……	……	
ON	ON	ON	ON	16

In this mode of operation, the trigger signal is an edge signal

(4) Startup method

- The multi-stage operation of the motor can be realized by a PLC/button to the corresponding IN pin with a level/edge start signal;
- Through the 485 communication, flip the “Polarity” bit in the corresponding IN pin configuration register to simulate an external IN trigger signal to realize multi-stage operation of the motor;

3.8 Internal Pulse Application Mode 21

The internal pulse application mode 21 is an analog speed control mode. The direction of the running direction can be set by an IN input start/stop signal, by IN or analog offset.

(1) Set an IN pin function: Input port setting register [60 to 65]

IN Pin	Settings	
	Polarity	Function bit
INx	0/1 (set according to input polarity)	7 (jog forward/start/stop: start and stop signal)
INx	0/1 (set according to input polarity)	8 (jog inversion / direction: direction signal)

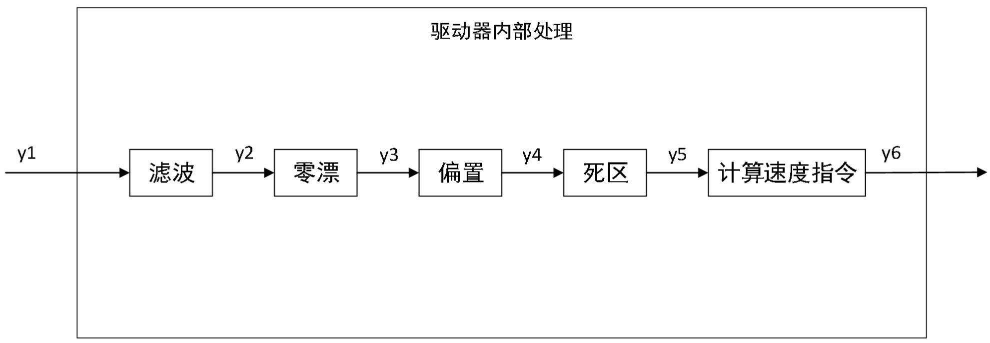
(2) Set the parameters of the analog input channel offset, filter, dead zone, zero drift, 3.3V corresponding speed: analog input parameter setting [272 ~ 279]

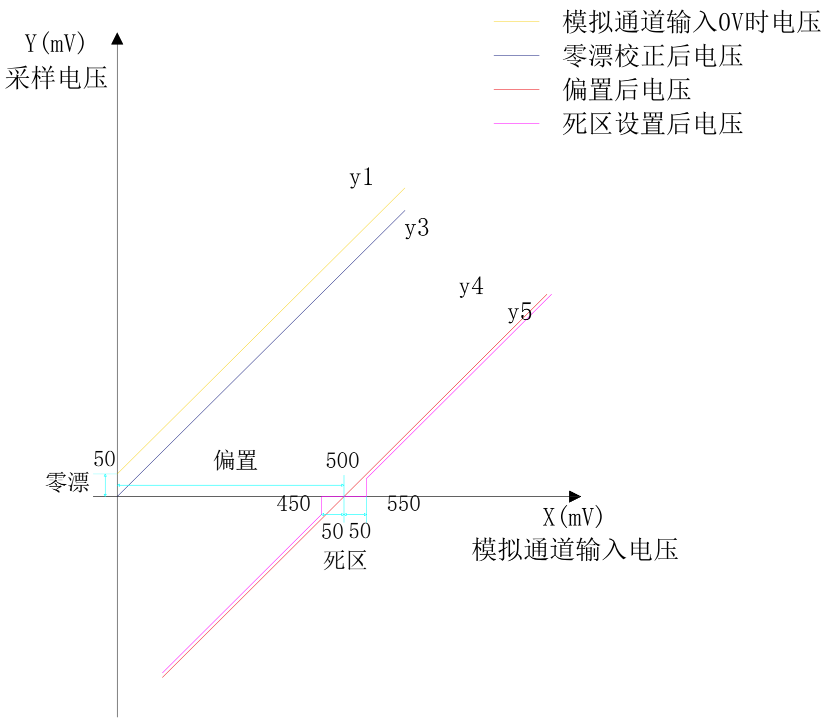
- Noun explanation

Zero drift: refers to the value of the drive sampled voltage value relative to GND when the analog channel input voltage is zero.

Offset: After the value is zero-drift corrected, the sampling voltage is zero and corresponds to the analog channel input voltage value.

Dead zone: refers to the input voltage range of the analog channel when the sampling voltage is zero.





- Filtering:

The driver provides analog channel filtering. By setting the low-pass filter cutoff frequency P273 parameter, it can prevent the motor command fluctuation caused by the unstable analog input voltage, and can also reduce the motor fault caused by the interference signal. The filtering function has no elimination or suppression of zero drift and dead zone.

- Zero drift correction:

When the actual input voltage is 0V, the analog channel output voltage deviates from the value of 0V.

In the figure, the analog channel output voltage that is not processed internally by the driver is shown as y1. Setting a large low-pass filter cutoff frequency assumes that the filtered sample voltage y2 is consistent with y1.

It can be seen that when the actual input voltage is x=0, the output voltage y1=50mV, at which time 50mV is called zero drift.

Manually set P275=50mV. After zero drift correction, the sampling voltage is as shown in y3. Y3=y1-50

By setting the offset and dead zone to 0, the zero drift value of the analog channel is calculated by the P277 parameter when the input is 0V.

- Offset setting:

When the sampling voltage is set to 0, the corresponding book input voltage value.

As shown in the figure, when the sampling voltage y4=0 is preset, the corresponding actual input voltage x=500mV, which is called offset.

Manually set P272=500mV, after biasing, the sampling voltage y4=x-500=y3-500

- Dead zone correction:

Limits the effective input voltage range when the drive sample voltage is not zero.

After the offset setting is completed, when the input voltage x is within 450mV and 550mV, the sampled voltage value is 0. This 50mV is called dead zone.

Set P274=50mV. After the dead zone is corrected, the sampling voltage is as shown in y5.

$$= \begin{cases} 0 & 450 \leq x \leq 550 \\ y_4 & 0 \leq x < 450 \text{ 或 } 550 < x \leq 3300 \end{cases}$$

- Calculate the speed command:

After the zero drift, offset, and dead zone setting is completed, the speed command value corresponding to 3.3V and the actual speed command y6 should be set by the P276 at this time.

Y_6=y_5/3300×P276

This value will be used as the speed command reference for the analog speed control mode.

After the correct setting is completed, the sampled voltage value of the analog input channel can be viewed in real time through P278, and the speed command value corresponding to the input analog quantity can also be viewed through P279.

(3) Startup method

- The analog speed operation mode of the motor can be realized by a PLC or a button to give a corresponding level start signal to the IN pin.
- Through 485 communication, flip the “Polarity” bit in the corresponding IN pin configuration register to simulate an external IN trigger signal, which can realize the

analog speed operation mode of the motor;

Note: The parameters of the analog channel offset, dead zone, zero drift, and 3.3V corresponding speed are valid after restart or direction switching.

3.9 Internal Pulse Application Mode 22

The internal pulse application mode 22 is an application mode of analog position following. There is no other IN start/stop or enable trigger signal, the position follows the absolute position operation mode, and the parameter setting of P84 is invalid.

(1) Set the offset, filter, dead zone and zero drift parameters of the analog input channel: analog input parameter setting [272~279]. For the meaning of the noun, please refer to the introduction of internal pulse application mode in the previous chapter. ;

(2) Set the position command corresponding to 3.3V: analog position control mode parameter [214~218];

(3) By changing the analog input voltage by means of a potentiometer or the like, the following operation of the analog position can be performed;

The position command value corresponding to the input analog voltage can be viewed through parameters P216/P217;

Since there is no external IN enable/start signal, the position of the analog input may be adjusted immediately after power-on, so beware of the resulting collision behavior!

Chapter 4 Appendix

Appendix A Function Code Message Format

Function 03 Read Hold Register:

Query message:

QUERY Field Name	Example(Hex)
Slave address	01
function code	03
Starting address high 8 bits	00
Starting address lower 8 bits	00
Data length high 8 bits	00
Data length lower 8 bits	05
CRC check lower 8 bits	85
CRC check high 8 bits	C9

Response message:

RESPONSE Field Name	Example(Hex)
Slave address	01
function code	03
Number of bytes returned	0A
High data (Register 40001)	00
Low data (Register 40001)	00
High data (Register 40002)	00
Low data (Register 40002)	01
High data (Register 40003)	00
Low data (Register 40003)	00

High data (Register 40004)	00
Low data (Register 40004)	03
High data (Register 40005)	FF
Low data (Register 40005)	FF
CRC check lower 8 bits	C5
CRC check high 8 bits	C6

Function 06 is written to a single register:

Query message: **01 06 00 12 00 00 29 CF**

QUERY Field Name	Example(Hex)
Slave address	01
function code	06
Address high 8 bits	00
Address lower 8 bits	12
Data high 8 bits	00
Data lower 8 bits	00
CRC check lower 8 bits	29
CRC check high 8 bits	CF

Response message:

QUERY Field Name	Example(Hex)
Slave address	01
function code	06
Address high 8 bits	00
Address lower 8 bits	12
Data high 8 bits	00
Data lower 8 bits	00
CRC check lower 8 bits	29

CRC check high 8 bits	CF

Function 16 (10 HEX) writes to multiple registers:

Query message: **01 10 00 4B 00 04 08 00 64 00 64 02 58 01 F4 86 EC**

QUERY Field Name	Example(Hex)
Slave address	**01**
function code	**10**
Starting address high 8 bits	**00**
Starting address lower 8 bits	**4B**
Data length high 8 bits	**00**
Data length lower 8 bits	**04**
Number of bytes	**08**
High data (Register 40076)	**00**
Low data (Register 40076)	**64**
High data (Register 40077)	**00**
Low data (Register 40077)	**64**
High data (Register 40078)	**02**
Low data (Register 40078)	**58**
High data (Register 40079)	**01**
Low data (Register 40079)	**F4**
CRC check lower 8 bits	**86**
CRC check high 8 bits	**EC**

Response message:

QUERY Field Name	Example(Hex)
Slave address	**01**
function code	**10**
Starting address high 8 bits	**00**

Starting address lower 8 bits	**4B**
Data length high 8 bits	**00**
Data length lower 8 bits	**04**
CRC check lower 8 bits	**B1**
CRC check high 8 bits	**DC**

Appendix B Modbus/RTU abnormal response and code

NT86 driver response and code when communication is not normal

```
// exception code
#define ILLEGAL_FUNCTION  0x01
#define ILLEGAL_DATA_ADD  0x02
#define ILLEGAL_DATA_VAL  0x03
#define DEVICEFAIL        0x04
```

Appendix C CRC Check

The cyclic redundancy check CRC area is 2 bytes, a 16-bit binary data. The CRC value is calculated by the transmitting device, and the calculated value is attached to the information. When receiving the information, the receiving device recalculates the CRC value, and compares the calculated value with the received actual value in the CRC area. If the two are not the same, Then an error is generated.

At the beginning of the CRC, first set the 16 bits of the register to "1", and then put the data of the adjacent two 8-bit bytes into the current one.

In the memory, only the 8-bit data of each character is used to generate the CRC, the start bit, the stop bit and the parity bit are not added to the CRC.

in. During the generation of the CRC, each 8-bit data is XORed with the value in the register, and the result is shifted to the right by one bit (to the LSB direction), and the MSB is filled with "0" to detect the LSB. If the LSB is "1", then The fixed value of the fixed value is XOR. If the LSB is "0", no exclusive OR operation is performed.

The above process is repeated until the shift is 8 times. After the 8th shift is completed, the next 8-bit data is XORed with the current value of the register. After all the information is processed, the final value in the register is the CRC value.

The process of generating CRC:

1. Set the 16-bit CRC register to FFFF.

2. The first 8-bit data is XORed with the lower 8 bits of the CRC register and the result is placed in the CRC register.

3. The CRC register is shifted one bit to the right, the MSB is filled with zeros, and the LSB is checked.

4. (If LSB is 0): Repeat 3 and move one bit to the right.

   (If LSB is 1): The CRC register is XORed with A001H

5. Repeat 3 and 4 until 8 shifts are completed, completing the processing of 8-bit bytes.

6. Repeat steps 2 through 5 to process the next 8-bit data until all bytes have been

processed.

7. The final value of the CRC register is the CRC value.

8. When placing the CRC value in the message, the upper 8 bits and the lower 8 bits should be placed separately. Put the CRC value in the message

Appendix D Modbus/RTU 16-bit CRC Check Routine

The CRC routines are written in C language specifications to facilitate porting to various platforms. The CRC_Checksum.c file contains two functions for calculating the CRC.

Try the CRC in a computational way:

```
unsigned short CalcCRCbyAlgorithm(unsigned char* pDataBuffer, unsigned long
usDataLen)
{
    /* Use the Modbus algorithm as detailed in the Watlow comms guide */

    const unsigned short POLYNOMIAL = 0xA001;
    unsigned short wCrc;
    int iByte, iBit;

    /* Initialize CRC */
    wCrc = 0xFFFF;

    for (iByte = 0; iByte < usDataLen; iByte++)
    {
        /* Exclusive-OR the byte with the CRC */
        wCrc ^= *(pDataBuffer + iByte);

        /* Loop through all 8 data bits */
```

```
        for (iBit = 0; iBit <= 7; iBit++)
        {
            /* If the LSB is 1, shift the CRC and XOR the polynomial mask with the CRC
*/

            /* Note - the bit test is performed before the rotation, so can't move the <<
here */
            if (wCrc & 0x0001)
            {
                wCrc >>= 1;
                wCrc ^= POLYNOMIAL;
            }
            else
            {
                /* Just rotate it */
                wCrc >>= 1;
            }
        }
    }

    return wCrc;
}
```

Calculate the CRC by looking up the table:

```
/* Table Of CRC Values */
const unsigned short TABLE_CRC16[] =
{
    0x0000, 0xC0C1, 0xC181, 0x0140, 0xC301, 0x03C0, 0x0280, 0xC241,
    0xC601, 0x06C0, 0x0780, 0xC741, 0x0500, 0xC5C1, 0xC481, 0x0440,
    0xCC01, 0x0CC0, 0x0D80, 0xCD41, 0x0F00, 0xCFC1, 0xCE81, 0x0E40,
```

```
    0x0A00, 0xCAC1, 0xCB81, 0x0B40, 0xC901, 0x09C0, 0x0880, 0xC841,
    0xD801, 0x18C0, 0x1980, 0xD941, 0x1B00, 0xDBC1, 0xDA81, 0x1A40,
    0x1E00, 0xDEC1, 0xDF81, 0x1F40, 0xDD01, 0x1DC0, 0x1C80, 0xDC41,
    0x1400, 0xD4C1, 0xD581, 0x1540, 0xD701, 0x17C0, 0x1680, 0xD641,
    0xD201, 0x12C0, 0x1380, 0xD341, 0x1100, 0xD1C1, 0xD081, 0x1040,
    0xF001, 0x30C0, 0x3180, 0xF141, 0x3300, 0xF3C1, 0xF281, 0x3240,
    0x3600, 0xF6C1, 0xF781, 0x3740, 0xF501, 0x35C0, 0x3480, 0xF441,
    0x3C00, 0xFCC1, 0xFD81, 0x3D40, 0xFF01, 0x3FC0, 0x3E80, 0xFE41,
    0xFA01, 0x3AC0, 0x3B80, 0xFB41, 0x3900, 0xF9C1, 0xF881, 0x3840,
    0x2800, 0xE8C1, 0xE981, 0x2940, 0xEB01, 0x2BC0, 0x2A80, 0xEA41,
    0xEE01, 0x2EC0, 0x2F80, 0xEF41, 0x2D00, 0xEDC1, 0xEC81, 0x2C40,
    0xE401, 0x24C0, 0x2580, 0xE541, 0x2700, 0xE7C1, 0xE681, 0x2640,
    0x2200, 0xE2C1, 0xE381, 0x2340, 0xE101, 0x21C0, 0x2080, 0xE041,
    0xA001, 0x60C0, 0x6180, 0xA141, 0x6300, 0xA3C1, 0xA281, 0x6240,
    0x6600, 0xA6C1, 0xA781, 0x6740, 0xA501, 0x65C0, 0x6480, 0xA441,
    0x6C00, 0xACC1, 0xAD81, 0x6D40, 0xAF01, 0x6FC0, 0x6E80, 0xAE41,
    0xAA01, 0x6AC0, 0x6B80, 0xAB41, 0x6900, 0xA9C1, 0xA881, 0x6840,
    0x7800, 0xB8C1, 0xB981, 0x7940, 0xBB01, 0x7BC0, 0x7A80, 0xBA41,
    0xBE01, 0x7EC0, 0x7F80, 0xBF41, 0x7D00, 0xBDC1, 0xBC81, 0x7C40,
    0xB401, 0x74C0, 0x7580, 0xB541, 0x7700, 0xB7C1, 0xB681, 0x7640,
    0x7200, 0xB2C1, 0xB381, 0x7340, 0xB101, 0x71C0, 0x7080, 0xB041,
    0x5000, 0x90C1, 0x9181, 0x5140, 0x9301, 0x53C0, 0x5280, 0x9241,
    0x9601, 0x56C0, 0x5780, 0x9741, 0x5500, 0x95C1, 0x9481, 0x5440,
    0x9C01, 0x5CC0, 0x5D80, 0x9D41, 0x5F00, 0x9FC1, 0x9E81, 0x5E40,
    0x5A00, 0x9AC1, 0x9B81, 0x5B40, 0x9901, 0x59C0, 0x5880, 0x9841,
    0x8801, 0x48C0, 0x4980, 0x8941, 0x4B00, 0x8BC1, 0x8A81, 0x4A40,
    0x4E00, 0x8EC1, 0x8F81, 0x4F40, 0x8D01, 0x4DC0, 0x4C80, 0x8C41,
    0x4400, 0x84C1, 0x8581, 0x4540, 0x8701, 0x47C0, 0x4680, 0x8641,
    0x8201, 0x42C0, 0x4380, 0x8341, 0x4100, 0x81C1, 0x8081, 0x4040
};
```

```
unsigned short CalcCRC_TAB(unsigned char* pDataBuffer, unsigned long
usDataLen)
{
    unsigned char nTemp;
    unsigned short wCRCWord = 0xFFFF;

    while (usDataLen--)
    {
        nTemp = wCRCWord ^ *(pDataBuffer++);
        wCRCWord >>= 8;
        wCRCWord ^= TABLE_CRC16[nTemp];
    }

    return wCRCWord;
}
```

CONTACT RTELLIGENT

Shenzhen Rtelligent Mechanical Electrical Technology Co., ltd

Address: B301,Zhuangbian Industrial Park, Nanchang Road, Gushu, Baoan District, Shenzhen, China

Tel: 86-755-29503086

Fax: 86-755-23327086

Website: www.szruitech.com

Email: marketing@szruitech.com

Office in East China

Address: Room 328, Building No.9, Shiyi Business Building, 5555 Shengzhuan Highway, Songjiang District, Shanghai

Contact: Mr. Ruan

Tel:1811749519

Mail: sales03@szruitech.com

Office in Shandong Province

Address: Room 601, Unit 1, Building No, 22, Xincaishi Street, Tianqiao District, Jinan, Shandong

Contact: Mr. Lu

Tel:13854109911

Mail: sales06 @szruitech.com