DAMIAO | 达妙科技

DM-JH11-2EC Geared Motor

User Manual V1.0 2025.05.19

Disclaimer

Thank you for purchasing the DAMIAO DM-JH11-2EC speed-reducing motor (hereinafter referred to as "the Motor"). Before using this product, please carefully read and comply with this document and all safety guidelines provided by DAMIAO DM-JH11-2EC. Failure to do so may cause harm to you or others, or damage the product or surrounding items. By using this product, you acknowledge that you have read, understood, and accepted all terms and contents of this document and any related materials. You agree to use the product solely for legitimate purposes and assume full responsibility for its use and any resulting consequences. DAMIAO DM-JH11-2EC shall not be liable for damages, injuries, or any legal liabilities arising from the direct or indirect use of this product.

Precautions

1. The motor must be operated strictly within the specified working environment and maximum allowable temperature range of the windings; otherwise, permanent irreversible damage may occur to the product.

2. Prevent foreign objects from entering the rotor, as this may cause abnormal operation.

3. Before use, check all components for integrity. If any parts are missing, aged, or damaged, stop using the device.

4. Ensure proper wiring and secure, correctly installed motor.

5. Do not touch the electronic rotor during operation to prevent accidents. The motor may overheat when delivering high torque, so take precautions to avoid burns.

6. Users must not disassemble the motor without authorization, as this may compromise its control accuracy or even cause malfunction.

7. To control the synchronization of position, speed, and torque mode using the host computer, you must use the new host computer version 2.0.3.6 or later.

8. Service Life Specifications: The service life of harmonic reducers is primarily determined by the wear of the wave generator bearings and flexible pulley, as calculated by the formula.

$$L_h = L_n \cdot \left(\frac{Tr}{T_{av}}\right)^3 \cdot \left(\frac{N_r}{N_{av}}\right)$$

The theoretical service life of the reducer can be calculated, where Tr is the rated torque of the reducer, Tav is the average load torque during actual operation, Nr is the rated speed of the reducer, Nav is the average speed during actual operation, and Ln is the design life of the reducer (theoretical value is 7000h). The impact of torque overload behavior on the service life of the reducer is very serious.

9. As a high-precision transmission component, the harmonic reducer's flexible pulley and flexible bearings are highly sensitive to load conditions. Any overload operation exceeding rated parameters may lead to accelerated fatigue and reduced lifespan. Ensure the harmonic joint module operates within its rated parameters. Damage to the flexible pulley or bearings caused by prolonged overload or unexpected shock conditions is not covered by our warranty.

Motor Features

1. Dual encoder with single-turn absolute position output, ensuring reliable position tracking even during power loss.

2. The motor and driver are integrated in a compact and highly modular design.

3. Enables visual debugging on the host computer and supports CAN parameter read/write.

4. Supports CAN FD with a maximum baud rate of 5Mbps

5. The CAN bus provides real-time feedback on motor parameters including speed, position, torque, and temperature.

6. Features dual-temperature protection.

7. Support CAN upgrade.

Specifications

Use the motor properly according to the following parameters.

	Rated voltage	24V (supports 24-48V)	
	Rated phase/power supply current	2.23A/1.19A@24V; 2.23A/0.54A@48V;	2.3A/1.0A@24V; 2.3A/0.5A@48V;

Motor Parameters	Peak phase / power supply current	4.26A/4.25A@24V； 4.26A/1.85A@48V；	3.6A/2.2A@24V； 3.6A/1.0A@48V；
	Rating torque	3.2NM	4.8NM
	Peak torque	7.8NM	10.5NM
	Rated speed	40rpm	20rpm
	Permissible maximum speed	60rpm	30rpm
Motor Characteristics	Reduction gear ratio	51: 1	101:1
	Number of pole-pairs	7	
	Phase inductance	538uH(@25℃)	
	Phase resistance	1.1Ω(@25℃)	
Structure & Weight	External diameter	52mm	
	Altitude	60mm	
	Mesopore size	8.5mm	
	Motor weight	Approximately 345 grams	
Encoder	Encoder bit count	17 bit	
	Number of encoders	2	
	Encoder type	Dual Encoders (Single-turn)	
Communication	Control interface	CAN@5Mbps(Max)	
	Parameter passing interface	UART@921600bps	
Control & Protection	Control model	MIT pattern	
		Speed mode	
		position mode	
		hybrid mode of power and position control	
		periodic synchronization mode	
		period synchronization mode	
		periodic synchronous torque mode	

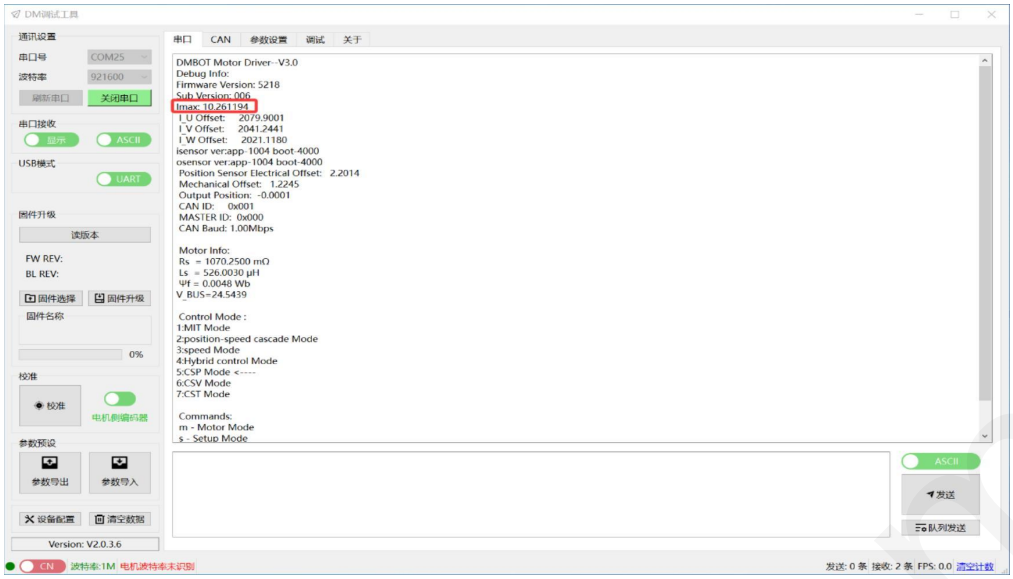
	Protections	Enable over-temperature protection with a protection temperature of 100°C. The motor will exit the enable mode if overheated.
		Motor over-temperature protection. Set according to usage requirements, recommended not to exceed 90°C. The motor will exit the 'enabled mode' if overheated.
		Motor overvoltage protection. Set according to usage requirements, recommended not to exceed 54V. Overvoltage will exit the "enabled mode".
		Communication loss protection. If no CAN commands are received within the set period, the system will automatically exit the enable mode.
		Motor overcurrent protection. Set according to usage requirements. It is recommended not to exceed 0.98. Overcurrent will exit the "enabled mode".
		Motor under-voltage protection: If the power supply voltage falls below the set value, the system exits the enable mode. The power supply voltage must not be less than 20V.

Operating Voltage

The operating voltage range is 24V-48V. Avoid hot plugging when exceeding 36V. The minimum operating voltage is 20V, and the maximum is 54V.

Maximum Phase Current

The maximum phase current of the corresponding drive can be queried by printing information through the serial port during power-on.

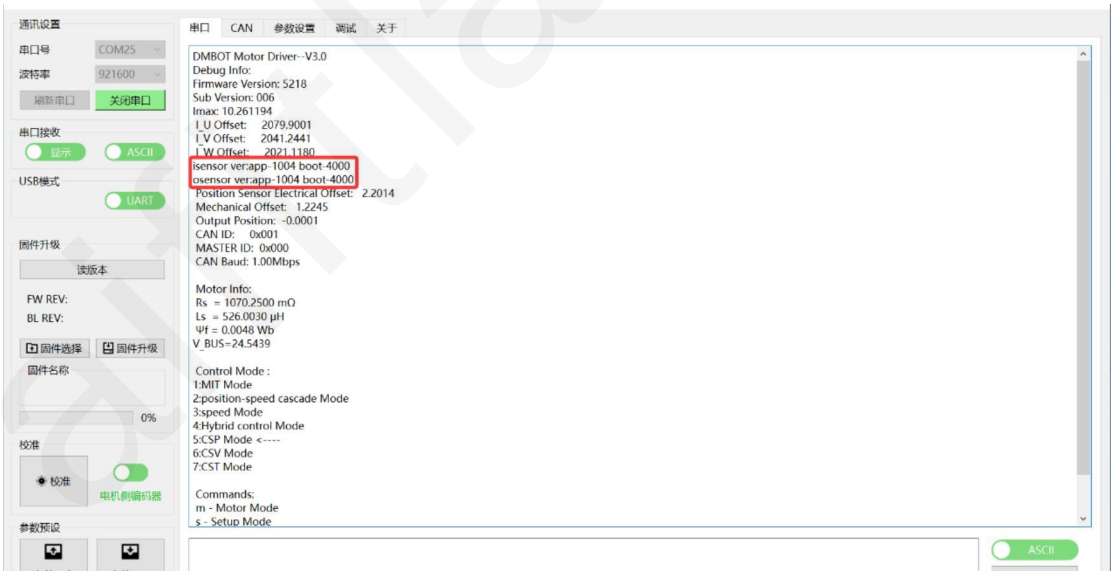


You can set the maximum phase current percentage to limit the operation through the debug assistant. The default value is 0.8, which is 80% of the maximum current that can be sampled. It is recommended that this value does not exceed 98%.

Encoder Version

The serial port can be used to print information during power-on to check the encoder version currently used by the motor, where 'isensor' refers to the motor rotor.

End encoder, osensor output shaft end encoder, app is the version number, boot refers to the bootloader serial number



Maximum Speed

The maximum rotational speed is constrained by multiple factors, including supply voltage (V_BUS), magnetic flux density (ψf), and gear ratio (GR). A typical upper limit can be calculated using the following formula, with the operational limit set at 2000rpm (rotor side).

$$V_{MAX}(rad/s) = 0.57735 * \frac{V_{BUS}}{N_{pp} \times GR \times \psi_f}(rad/s)$$

In the formula,

VBUS is the power supply voltage

Npp is the number of motor pole pairs

ψf represents the rotor magnetic flux

Torque Constant

The torque coefficient of the motor can be considered constant within its rated range. When combined with the gearbox, the calculation can be performed using the following formula:

$$K_t = 1.5 * N_{pp} * \psi_f * GR * GREF$$

where,

Npp: Pole pairs

Ψf: Rotor flux linkage

GR: Motor reduction ratio

GREF: Gearbox torque transmission coefficient

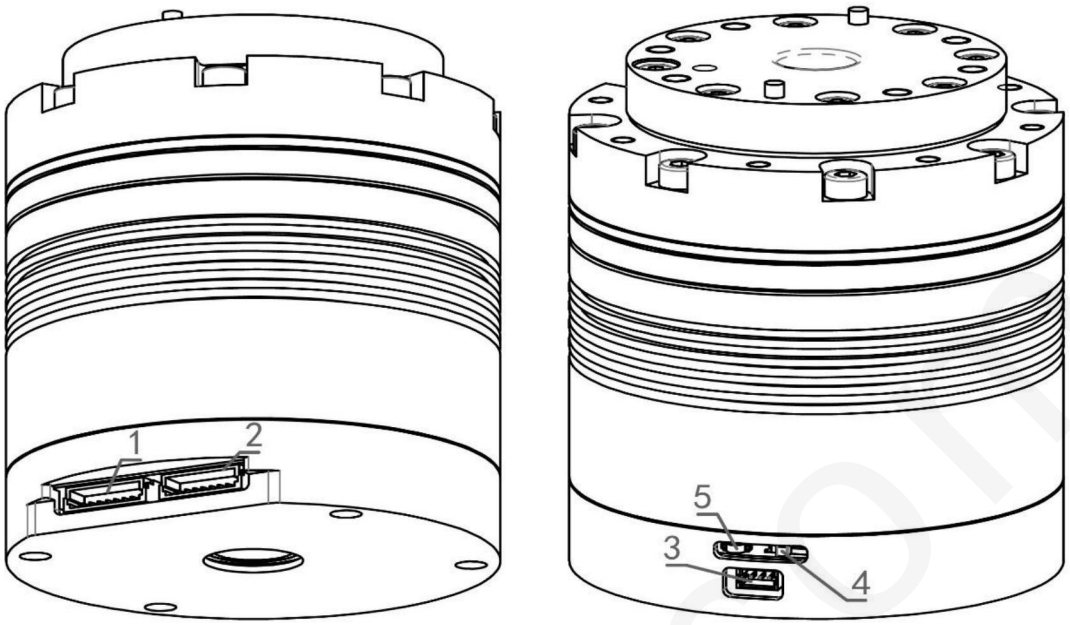
Package Contents

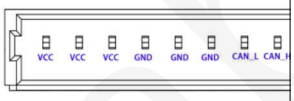
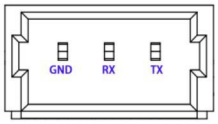
1. Motor (including drive) × 1

2. Power supply (including CAN communication terminals) connection cable: SH1.08-pin cable (200mm) ×1

3. Debugging the serial port signal line: SH1.0 connection cable-3-pin (200mm) ×1

Interface and Wire Sequence Description



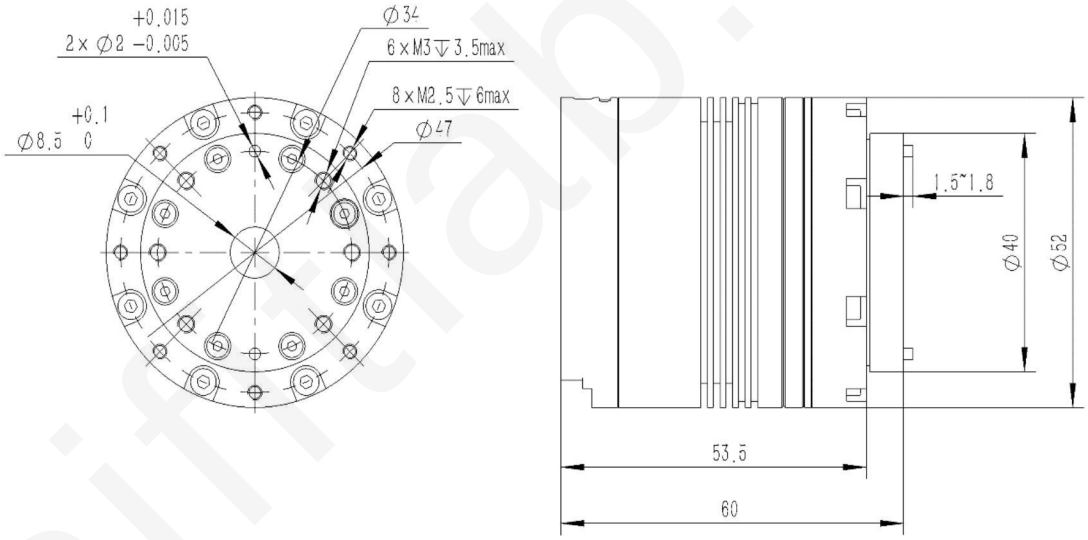
Specific Name-Number	Interface annotation	explain
Power Supply + CAN Communication Terminal （1）（2）		1. Connect the power supply via the SH1.0 8-pin cable, rated at 24V (operating within 24-48V range) to power the motor. 2. The system connects to external control devices via CAN communication terminals, receiving CAN control commands and transmitting motor status data.
Debug serial port (3)		Connect the SH1.0 connector (3-pin) to the intermediate conversion board (SH1.0 3-pin + 8-pin to XT30 + GH1.25), then use the USB-to-CAN diagnostic tool (or a universal USB-to-serial module) to connect to the PC. Use Damiao Technology's diagnostic assistant to configure motor parameters and perform firmware upgrades.
motor indicator light （5）	not have	Red light always on: The motor is in an inactive state Red light always on: The motor is enabled Red light flashing: Motor error. Check the feedback frame for details.

		Green light flashing: Power-on self -test failed. Take a screenshot of the power-on printout and contact customer service.
terminal resistance switch（4）		The terminal resistance of the motor configuration is enabled by default.

Note: When connecting the cable to the motor port, ensure the terminals are correctly oriented to prevent pin misalignment or damage.

Motor Dimensions and Installation

Refer to the motor mounting hole dimensions and positions to install the motor on the corresponding equipment.



Indicator Status

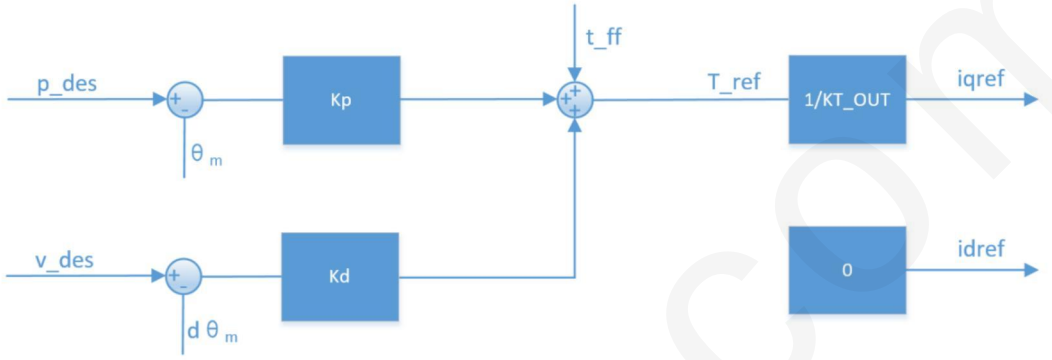
Normal Status	The green light is always on.	The ERR bit is 1, indicating the enabled mode and normal operating state.

	Red light always on	The ERR bit is 0, indicating disabled mode
Abnormal Status	Red LED flashing	Indicates a fault. The corresponding fault type is: 2 — Motor-side encoder not recognized 3 — Output shaft encoder not recognized 5 — Encoder reading error 8-over pressure; 9-under pressure; 8-over pressure; 9-under pressure; A—Overcurrent; B—MOS Overtemperature; C-Motor coil overheating; D-Communication loss; E-Overload; (You can check the fault through the feedback frame or the Dami Technology Debug Assistant interface)

Operating Modes

MIT Mode

The MIT mode is designed to maintain compatibility with the original MIT mode, enabling seamless switching while allowing flexible configuration of control parameters (P_MAX, V_MAX, T_MAX). The electronic controller converts received CAN data into control variables, calculates torque values, and supplies them as current references for the current loop. The current loop then adjusts according to its regulation rules to achieve the target torque current. (This motor incorporates a new speed limit to prevent exceeding the maximum allowable speed.) The control block diagram is shown below:

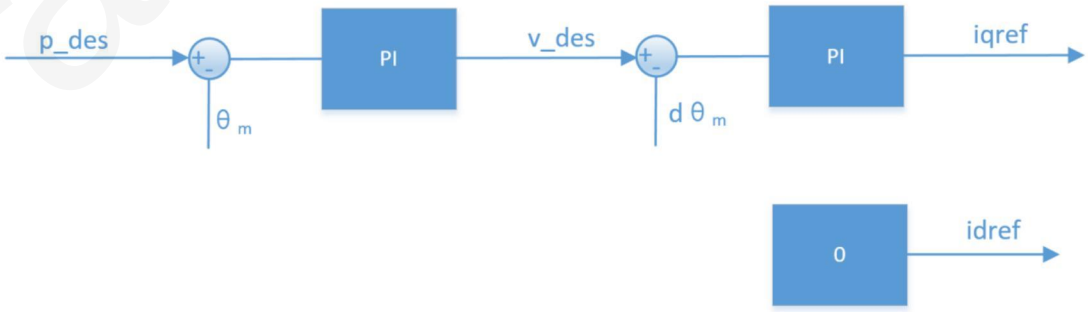


The MIT model can be adapted to various control configurations. For instance, when kp=0 and kd≠0, a constant-speed rotation can be achieved by setting v_des. Similarly, when kp=0 and kd=0, torque output can be controlled by setting t_ff.

Note: When controlling position, do not set kd to 0, as this may cause motor oscillation or even loss of control.

Position-Speed Mode

The position cascade control mode employs a three-loop series configuration. The outermost position loop provides the speed loop's setpoint, while the inner current loop receives the speed loop's output to regulate actual current. The control block diagram is shown below:



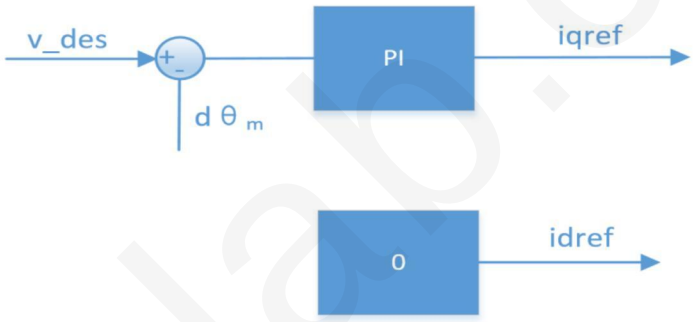
p_des is the target position for control, while v_des defines the maximum absolute speed during motion.

When the cascade mode is controlled using the parameters recommended by the debugging assistant, it achieves better control accuracy and a smoother process, though with a longer response time. In addition to v_des, other configurable parameters include acceleration/deceleration settings. If additional oscillations occur during control, increasing the acceleration/deceleration can mitigate them.

Note: p_des and v_des are measured in rad and rad/s respectively, with data type float. The damping factor must be set to a positive non-zero value. Refer to the speed mode precautions for details.

Speed mode

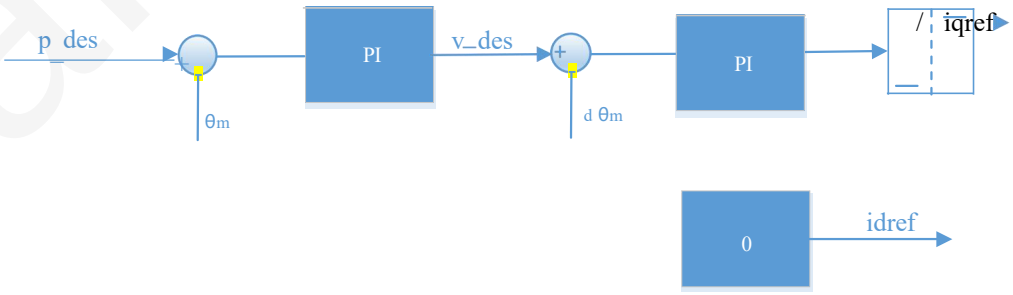
The speed mode enables the motor to operate stably at the set speed, as shown in the control block diagram below.



Note: v_des is measured in rad/s and is a float data type.

Torque-Position Hybrid Mode

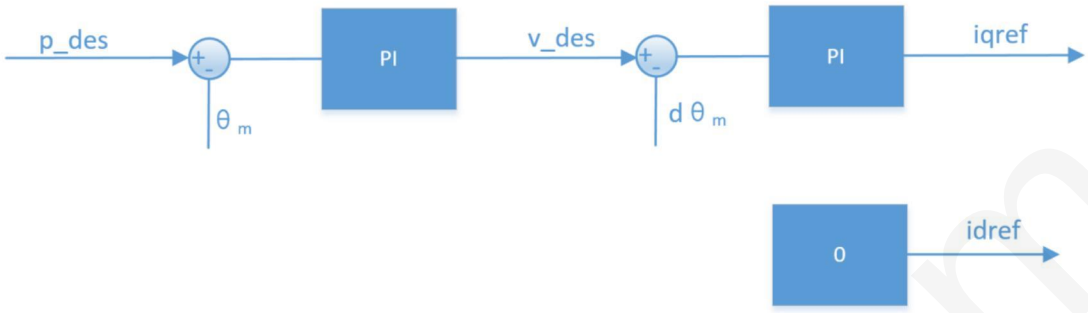
The position-force hybrid control mode dynamically adjusts the output torque based on position-speed mode control, as shown in the control block diagram below:



The current loop is limited to a given range by adding a current command saturation link after the output command of the speed loop.

Cyclic Synchronization Positioning (CSP)

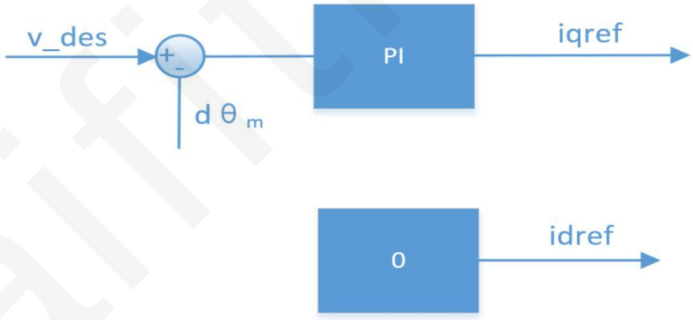
In this mode, the absolute position preset values are transmitted to the controller via the fieldbus at fixed intervals (referred to as "cycles"). The controller then stops calculating the ramp and operates solely based on the preset values. The control block diagram is shown below:



In this mode, position commands should be issued progressively rather than abruptly, as abrupt commands may cause motor vibration (this mode limits input parameters). The upper level should first plan the curve and then deliver the planned curve to the motor step by step.

Cycle Synchronization Speed Mode (CSV)

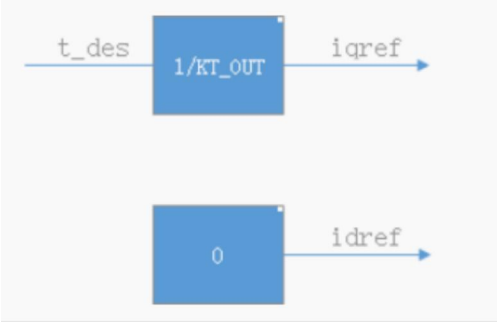
In the periodic synchronization speed mode, the host controller periodically transmits the calculated target speed to the servo driver, with speed and torque adjustments performed internally by the servo driver. The control block diagram is shown below:



In this mode, position commands should be issued progressively rather than abruptly, as abrupt commands may cause motor vibration (this mode limits input parameters). The upper level should first plan the curve and then deliver the planned curve to the motor step by step.

CST (Clock Synchronization Torque)

In the periodic synchronous torque mode, the host controller periodically synchronizes the calculated target torque to the servo driver, which executes torque regulation internally. When the speed reaches the limit value, the system enters the speed regulation phase. The control block diagram is shown below:



In this mode, torque commands should be issued progressively rather than abruptly, as abrupt commands may cause motor vibration (this mode imposes limits on input parameters). The upper-level system should first plan the curve and then deliver the planned curve to the motor step by step.

Mode Modification

The mode can be set via the host computer's serial port. Simply select the desired mode and click' Write Parameters'. After successful configuration, the motor will reset automatically. The mode is stored in the motor driver and remains intact upon power-on.

Additionally, modifications can be made via the CAN interface to update the mode register. For details, refer to the next chapter.

In the "Mode Switch" section, this setting prevents the motor from resetting but resets the following five variables to zero:

1. Location command value

2. Speed command value

3. Torque command value (MIT mode)

4.kp（MIT pattern）

5.kd（MIT pattern）

The mode will not be saved without the "Save parameters" command. It will be lost after a power outage and will be restored to the last saved mode upon power-on.

CAN communication

After completing motor calibration, parameter setting, and configuration, the system becomes operational. It uses CAN standard frame (STD) format with a default baud rate of 1Mbps, though command-based switching to different baud rates is available (see CAN baud rate modification section). Functionally, frames are categorized into receiving frames and feedback frames. Receiving frames contain control data for motor command execution, while feedback frames transmit status data to the upper controller. The feedback mechanism operates on an inquiry basis: when the driver receives a frame ID matching the motor's configured CAN ID (with low 8 bits for parity check and high 3 bits ignored), it sends the current status data to the bus. Although receiving frame formats and IDs vary by motor mode, all feedback frames maintain identical structure across all operational modes.

Change baud rate

The baud rate can be configured via the host computer's serial port. Simply select the desired baud rate and click' Write Parameters'. After successful setup, the motor will automatically reset, with the baud rate stored in the motor driver for permanent retention during power cycles.

Another method is to modify the settings via the CAN interface by writing the baud rate register value, as detailed in the 'CAN Baud Rate Modification' section of this chapter.

Note: Modifying the baud rate via CAN may fail if multiple devices are connected to the bus. Exercise caution and configure the baud rate before installation.

feedback frame

The feedback frame ID is set by the debug assistant (Master ID), with a default value of 0. It primarily provides feedback on the motor's position, speed, and torque. The frame format is defined as:

feedback message	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
MST_ID	ID|ERR<<4	POS[15:8]	POS[7:0]	VEL[11:4]	VEL[3:0]|T[11:8]	T[7:0]	T_MOS	T_Rotor

among：

The ID indicates the controller's identifier, with the lower 8 bits of CAN_ID (ERR) representing the status, corresponding to the following status types:

0-Disability;

1—Enable;

2-The motor-side encoder is not recognized;

3-The output shaft encoder is not recognized;

5-Reading encoder error;

8—overpressure;

9-Under-voltage;

A—— excess current ；

B—MOS overtemperature;

C-Motor coil overheating;

D-Communication loss;

E—— overload ；

POS indicates the motor's position information

VEL indicates the motor speed information

T represents the torque information of the motor

T_MOS is the average temperature of the upper MOS, in ℃

T_Rotor is the average temperature of the motor's internal coils, measured in ℃

The system converts floating-point data to signed fixed-point data using linear mappings for position, speed, and torque. Specifically, position is represented by 16-bit data, while speed and torque are both 12-bit, as shown in the figure below.

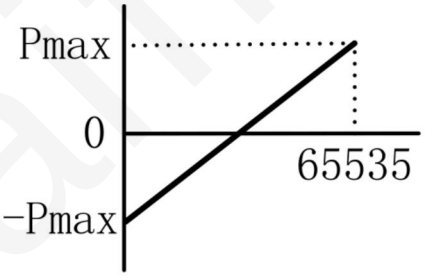


Figure: Position Linear Mapping

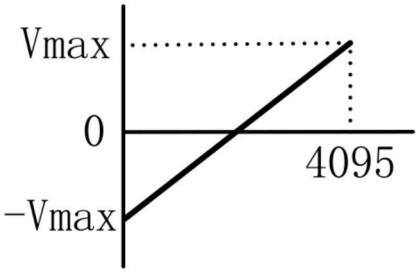


Figure: Speed Linear Mapping Diagram

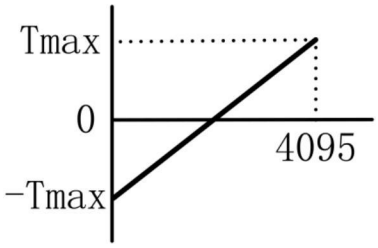


Figure: Torque Linear Mapping Diagram

[Note]:

①After power-on, the motor's position output is limited to [-π, π] radians.

②The position is measured in radians (rad) and indicates the output shaft's position after deceleration. All subsequent position descriptions shall beThis definition will not be repeated.

③ The unit of speed is rad/s (radians per second), representing the output shaft's speed after deceleration. All subsequent descriptions of speed shall follow this definition and need not be repeated.

④ Torque is measured in Nm and refers to the torque output by the shaft, specifically the torque after reduction. All subsequent descriptions of torque shall follow this definition and need not be repeated.

Control frame in MIT mode

control message	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
ID	p_des [15:8]	p_des [7:0]	v_des [11:4]	v_des[3:0]| Kp[11:8]	Kp [7:0]	Kd [11:4]	Kd[3:0]| t_ff[11:8]	t_ff[7:0]

The frame ID equals the set CAN ID value

P_des: Position given

V_des: Speed given

Kp: Position Ratio Coefficient

Kd: Positional differential coefficient

T_ff: Torque setpoint

All parameters adhere to the mapping relationships specified in the previous section. The values of p_des, v_des, and t_ff can be configured via the debugging assistant, while Kp ranges from 0 to 500 and Kd ranges from 0 to 5.

A standard CAN data frame is 8 bytes long. MIT's control command format encodes five parameters—Position, Velocity, Kp, Kd, and Torque—into 8 bytes using bit combinations. Specifically, Position occupies 2 bytes (16 bits), while Velocity, Kp, and Kd each occupy 12 bits.

A standard CAN data frame is 8 bytes long. MIT's control command format encodes five parameters—Position, Velocity, Kp, Kd, and Torque—into 8 bytes using bit combinations. Specifically, Position occupies 2 bytes (16 bits), while Velocity, Kp, and Kd each occupy 12 bits.

The MIT command converts floating-point data to integers proportionally and sends them to the driver, which then converts the received integers back to floating-point data proportionally. This conversion is handled by two functions, uint_to_float and float_to_uint, as detailed below:

```
1.  float uint_to_float(int x_int, float x_min, float x_max, int bits){
2.  /// converts unsigned int to float, given range and number of bits ///
3.  float span = x_max - x_min;
4.  float offset = x_min;
5.  return ((float)x_int)*span/((float)((1<<bits)-1)) + offset;
6.      }
```

```
1.  int float_to_uint(float x, float x_min, float x_max, int bits){
2.      /// Converts a float to an unsigned int, given range and number of bits ///
3.      float span = x_max - x_min;
4.      float offset = x_min;
5.      return (int) ((x-offset)*((float)((1<<bits)-1))/span);
6.      }
```

To apply these two conversion functions, first determine the maximum and minimum values for proportional scaling. These parameters can be found in the parameter settings page, where KP and KD default to 0.0~500.0 and 0.0~5.0 respectively. The parameters Pos, Vel, and Torque are preset to ±12.5, ±10, and ±12 as shown in the figure below. These values can be adjusted according to the motor's actual parameters. However, when sending control commands, they must match the set values to prevent proportional scaling.

Control frame in position velocity mode

control message	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
0x100+ID	p_des				v_des			

The frame ID is the set CAN ID value plus an offset of 0x100.

P_des: Position given, floating-point type, with the lower bit first and the higher bit last

V_des: Speed given, floating-point, lower bits first, higher bits last

The CAN ID for the command is 0x100+ID. The speed setting is the maximum speed value during operation.

Control frame in speed mode

control message	D[0]	D[1]	D[2]	D[3]
0x200+ID	v_des			

The frame ID is the set CAN ID value plus an offset of 0x200.

V_des: A floating-point variable with fixed speed, where the lower bit is first and the higher bit is last. The CAN ID for this command is 0x200+ID.

Control frame in position and force hybrid control mode

control message	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
0x300+ID	p_des				v_des		i_des	

P_des: Position given, in radians, floating-point type with least significant bit first and most significant bit last;

V_des: Speed limit value in radians per second (rad/s), scaled by 100x,16-bit unsigned (LSB first, MSB last), range 0-10000 (clamped to 10000). The actual speed limit is 0-100 rad/s.

I_des: The torque current limit is set to a 10,000x magnitude, using an unsigned 16-bit type with the lower bit first and the higher bit last. The range is 0-10,000, and it is capped at 10,000 if exceeded. The corresponding actual current limit is 0-1.0.

Current magnitude: The actual current value divided by the maximum phase current value.

Control Frame in Synchronous Periodic Speed Position Mode

control message	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
0x400+ID	p_des				v_des			

The frame ID is the set CAN ID value plus the offset of 0x400.

p_des: Position given, floating-point type, with the lower bit first and the higher bit last

v_des: speed given, floating-point, lower bits first, higher bits last

The CAN ID for the command is 0x400+ID. The speed setting is the maximum speed value during operation.

Control frame in synchronization mode

control message	D[0]	D[1]	D[2]	D[3]
0x500+ID	v_des			

The frame ID is the set CAN ID value plus the offset of 0x500.

v_des: A floating-point variable with fixed speed, where the lower bit is first and the higher bit last. The CAN ID for this command is 0x500+ID.

Control Frame in Synchronous Torque Mode

control message	D[0]	D[1]	D[2]	D[3]
0x600+ID	t_des			

The frame ID is the set CAN ID value plus the offset of 0x600.

t_des: torque given, floating-point, lower bit first, higher bit last

The CAN ID for the command is 0x600+ID.

Enable

After the power-on self-test completes, the 'enable' command must be sent to initiate control. The 'enable' frame is a control frame. As mentioned earlier, the frame

ID differs, but the data segment remains consistent. Regardless of the mode, the 'enable' data definition is identical, as shown below:

D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFC

Disability

Disabling is the default state for motor power-on, where all three-phase terminal voltages are identical, each being a 50% modulated wave of the supply voltage. The 'disabling' frame is a control frame, with the frame ID as previously defined and the data segment defined as follows:

D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFD

Save position zero

The "Save Position Zero Point" frame is a control frame that resets the current output axis to zero position and sets the position value to 0. As mentioned earlier, the frame ID and data segment definition are as follows:

D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFE

Clear errors

When the motor experiences overheating or other faults, the 'Clear' command can be sent to resolve the issue. The 'Clear' frame is a control frame, with the frame ID as previously described. The data segment is defined as follows:

D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFB

Read parameters

message ID	attribute	D[0]	D[1]	D[2]	D[3]
0x7FF	STD	CANID_L	CANID_H	0x33	RID

RID is the register address, see Appendix <Register List and Range>

After successful read, the data of the register is returned, with the frame format as follows:

message ID	attribute	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
MST_ID	STD	CANID_L	CANID_H	0x33	RID	data			

The data is either floating-point or unsigned integer, occupying 32 bits (4 bytes), with D4 as the least significant bit and D7 as the most significant bit, and so on.

In parameter

message ID	attribute	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
0x7FF	STD	CANID_L	CANID_H	0x55	RID	data			

As described above, the RID returns the written data upon successful completion, with the frame format identical to the original sent data.

message ID	attribute	D[0]	D[1]	D[2]	D[3]	D[4]	D[5]	D[6]	D[7]
MST_ID	STD	CANID_L	CANID_H	0x55	RID	data			

Writing register data takes effect immediately but cannot be stored. The data is lost after power failure. To write all modified parameters into the chip, a command to send storage parameters must be sent.

Storage parameters

message ID	attribute	D[0]	D[1]	D[2]	D[3]
0x7FF	STD	CANID_L	CANID_H	0xAA	0x01

After successful write, the return format is:

message ID	attribute	D[0]	D[1]	D[2]	D[3]
MST_ID	STD	CANID_L	CANID_H	0xAA	0x01

[Note]:

The storage parameters are only effective in the disabled mode.

② All parameters will be saved at once when storing.

③ This operation writes parameters to the on-chip flash memory, with each operation taking up to 30ms. Ensure you allow sufficient time.

④ The flash memory can withstand approximately 10,000 erase cycles. Avoid frequent transmission of the "storage parameters" command.

Mode switch

Supports switching between multiple modes. The available control modes are:

Code	Mode
1	MIT
2	Velocity of position
3	Velocity
4	Hybrid control of position and force
5	Synchronization position
6	Synchronization speed
7	Synchronization speed

The mode can be modified by changing the value of the mode register (0x0A). When switching modes, the motor first resets all command values, including position, speed, and the torque feedforward and KP/KD values in MIT mode.

To prevent shock during mode switching from position control to another mode, it is recommended to first read the precise position (value of register 0x50) before considering the switch, and to perform the switch when the motor is at zero speed.

The modified mode will not be saved to flash memory and will be lost after power loss. Upon power restoration, the control mode will revert to the last saved configuration in flash.

CAN baud rate modification

The current CAN communication baud rate can be modified by writing specific data to the baud rate register (address 0x23), supporting customized baud rate settings. The supported baud rates are as follows:

Code	Baud rate
0	125K
1	200K
2	250K
3	500K
4	1M
5	2M
6	2.5M
7	3.2M
8	4M
9	5M

After successfully modifying the baud rate, the driver first transmits data at the original baud rate before switching to the new rate. Upon power-on, the motor first checks the stored baud rate. If it exceeds 5Mbps, the system automatically defaults to 1Mbps. For rates above 1Mbps (excluding 1Mbps), the motor switches to CAN FD mode. If the baud rate is ≤1Mbps, it automatically switches to CAN 2.0B. Motors configured as CAN FD can still receive CAN 2.0B data frames but use CAN FD for feedback transmission, causing the upper-layer controller to miss feedback data and triggering continuous driver error reports. Controllers using CAN 2.0B can revert to the original baud rate by issuing a baud rate modification command after missetting the ID.

Register list and range

address （HEX）	address （DEC）	variable	description	read-write	scope	type
0x00	0	UV_Value	low voltage protection value	RW	(10.0,fmax]	float
0x01	1	KT_Value	torque coefficient	RW	[0.0,fmax]	float
0x02	2	OT_Value	over temperature protection value	RW	[80.0,200)	float
0x03	3	OC_Value	overcurrent protection value	RW	(0.0,1.0)	float
0x04	4	ACC	accelerated speed	RW	(0.0,fmax)	float
0x05	5	DEC	deceleration	RW	[-fmax,0.0)	float
0x06	6	MAX_SPD	maximum speed	RW	(0.0,fmax]	float
0x07	7	MST_ID	feedback ID	RW	[0,0x7FF]	uint32
0x08	8	ESC_ID	receive ID	RW	[0,0x7FF]	uint32
0x09	9	TIMEOUT	Timeout alert time	RW	[0,2^32-1]	uint32
0x0A	10	CTRL_MODE	control model	RW	[0,7]	uint32
0x0B	11	Damp	motor viscosity coefficient	RO	/	float
0x0C	12	Inertia	rotating inertia of motor	RO	/	float
0x0D	13	hw_ver	continue to have	RO	/	uint32
0x0E	14	sw_ver	Software version	RO	/	uint32
0x0F	15	SN	continue to have	RO	/	uint32
0x10	16	NPP	number of pole pairs of motor	RO	/	uint32
0x11	17	Rs	motor phase resistance	RO	/	float
0x12	18	Ls	motor phase inductance	RO	/	float
0x13	19	Flux	motor flux linkage	RO	/	float
0x14	20	Gr	gear reduction ratio	RO	/	float
0x15	21	PMAX	position mapping range	RW	(0.0,fmax]	float
0x16	22	VMAX	velocity mapping range	RW	(0.0,fmax]	float
0x17	23	TMAX	torque mapping range	RW	(0.0,fmax]	float
0x18	24	I_BW	current loop control bandwidth	RW	[100.0,1.0e4]	float
0x19	25	KP_ASR	speed ring Kp	RW	[0.0,fmax]	float
0x1A	26	KI_ASR	speed ring Ki	RW	[0.0,fmax]	float
0x1B	27	KP_APR	position loop Kp	RW	[0.0,fmax]	float

0x1C	28	KI_APR	position loop Ki	RW	[0.0,fmax]	float
0x1D	29	OV_Value	overvoltage protection value	RW	TBD	float
0x1E	30	GREF	gear torque efficiency	RW	(0.0,1.0]	float
0x1F	31	Deta	speed loop damping coefficient	RW	[1.0,30.0]	float
0x20	32	V_BW	speed loop filter bandwidth	RW	(0.0,500.0)	float
0x21	33	IQ_c1	current loop enhancement factor	RW	[100.0, 1.0e4]	float
0x22	34	VL_c1	speed loop enhancement factor	RW	(0.0,1.0e4]	float
0x23	35	can_br	CAN baud rate code	RW	[0,4]	uint32
0x25	37	Boot_ver	Boot version number	RO	/	uint32
0x36	54	x_off	Output shaft angle offset	RO	/	float
0x37	55	dir	direction	RO	/	float
0x38	56	m_off	motor side angular offset	RO	/	float
0x3C	60	VBus	supply voltage	RO	/	Float
0x50	80	p_m	Motor current position	RO	/	float
0x51	81	xout	Output shaft position	RO	/	float

RW: Read-write.

RO： read only .

Motor Debugging Procedure

Use a host computer version V2.0.3.6 or later for debugging. The following demonstrates the debugging process with V2.0.3.6.

Connect the Device

First, connect the motor's serial port, CAN port, and power interface. On the computer, launch the debugging assistant software and select the appropriate settings.

Open the serial port.

When power is supplied to the motor, the serial port will print the following information. Control Mode: Indicates the current drive mode. Different modes

Use different command formats (see CAN Communication section).

Encoder Calibration

The primary purpose of motor encoder calibration is to correct sensor installation errors. During calibration, the motor rotates one full rotor cycle in both directions. Ensure the motor rotates freely and operates under no-load conditions to prevent calibration

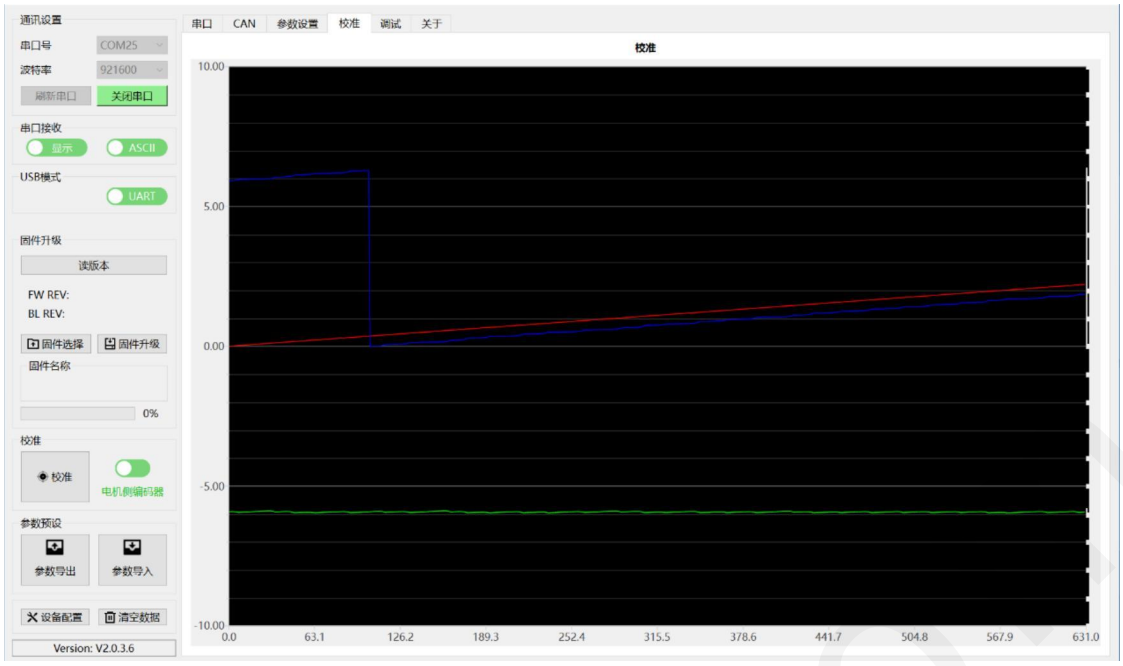
failure. The motor has undergone factory calibration and has internal calibration data. Unless abnormal conditions occur, the motor can be used directly without additional calibration. Below are the calibration procedures. For abnormal conditions such as driver board replacement or unusual vibrations, calibration should be performed before operation.

Step 1: Motor-side encoder calibration

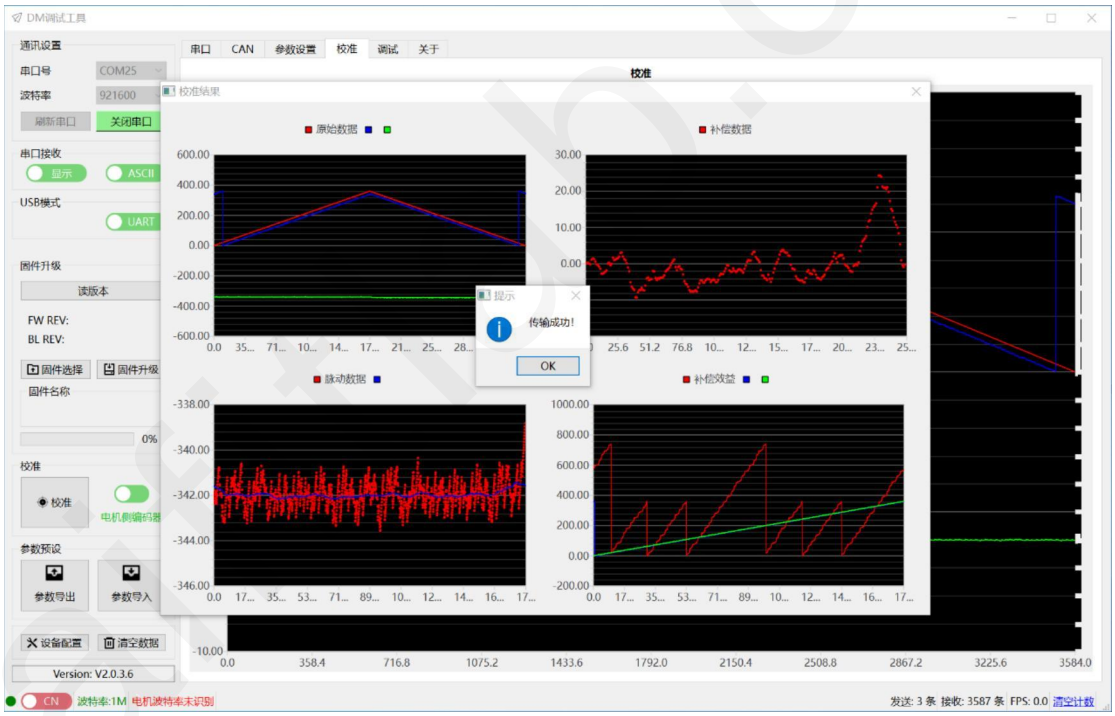
Click Calibrate. The motor will rotate first, and the pole pairs will be returned to the debugging assistant after rotation.

During this process, the motor will rotate. Make sure to secure it properly to prevent accidents.

After completing the motor pole pair identification, the motor will automatically calibrate the sensor and display its waveform diagram, as shown below:



Step 2: After calibration is completed, the values will be automatically uploaded. For example:



Parameter Calibration

Parameter identification primarily determines critical motor parameters including phase resistance, phase inductance, and magnetic flux. The motor is factory-calibrated and pre-programmed, enabling direct operation without recalibration. Below are the calibration procedures:

Click the "Parameter Settings" tab, then click the "Parameter Calibration" button to initiate the driver's identification process. During this time, the motor will rotate; ensure it remains unloaded and securely fastened.

After recognition, the results will be uploaded automatically.

The viscosity coefficient is only for reference and can be calibrated many times.

Note: All parameters except the viscosity coefficient must be positive. If a negative value occurs, verify the motor status before labeling.

Output shaft encoder calibration

The dual encoder motor requires output shaft encoder calibration to enhance accuracy. During calibration, the motor must complete one full forward rotation. Ensure the motor rotates freely and operates under no-load conditions to prevent excessive calibration errors. The motor comes with factory-calibrated output shaft encoders pre-programmed internally. Under normal circumstances, these encoders can be used directly without modification.

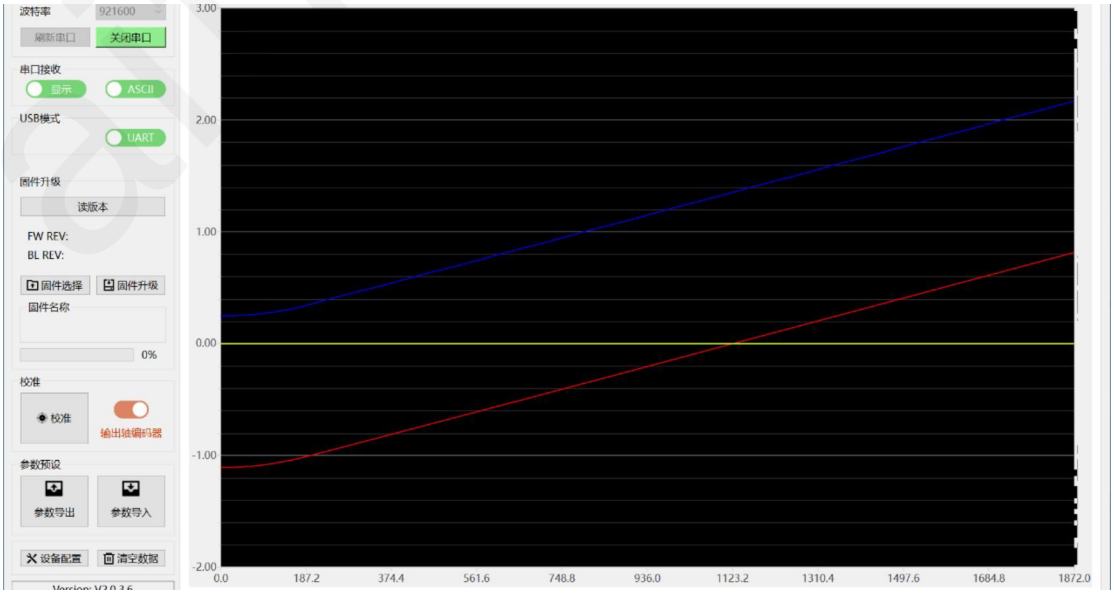
No further output shaft encoder calibration is required. The following provides the operational steps for output shaft encoder calibration, which should be performed after the motor's drive board is replaced.

For abnormal conditions such as output position changes, calibrate the output shaft encoder before operation.

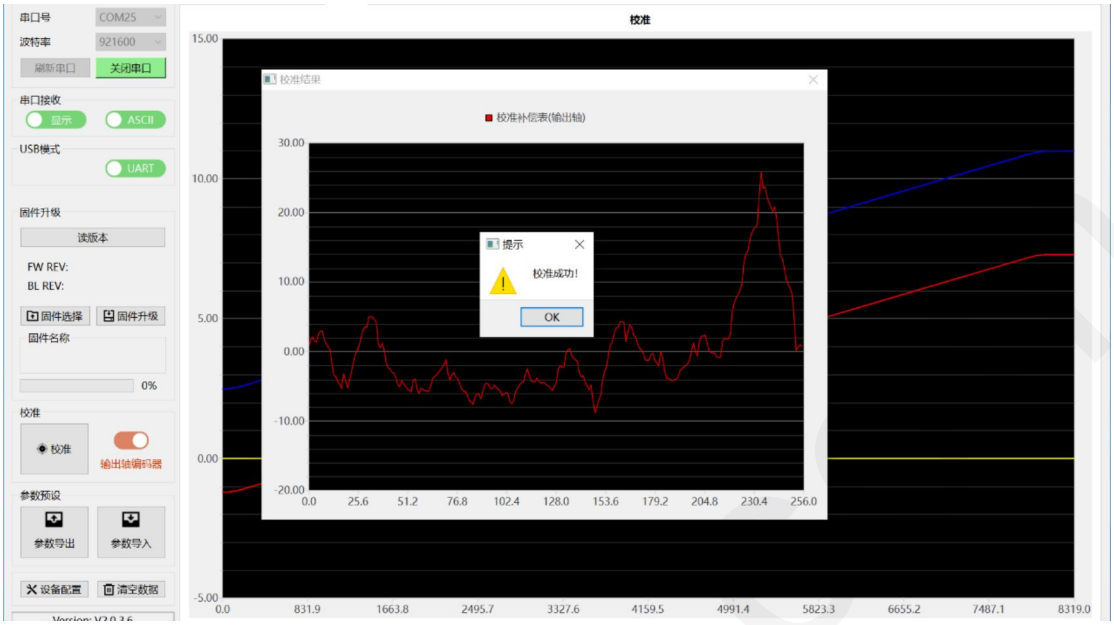
Step 1: Output shaft encoder calibration

Click the slider next to the calibration button to display the "Output Shaft Encoder" status, then click Calibrate

The motor starts rotating and uploads the encoder's raw data waveform, as shown in the figure below.



Step 2: After calibration is completed, the system will automatically upload and display the calibration value as shown below:



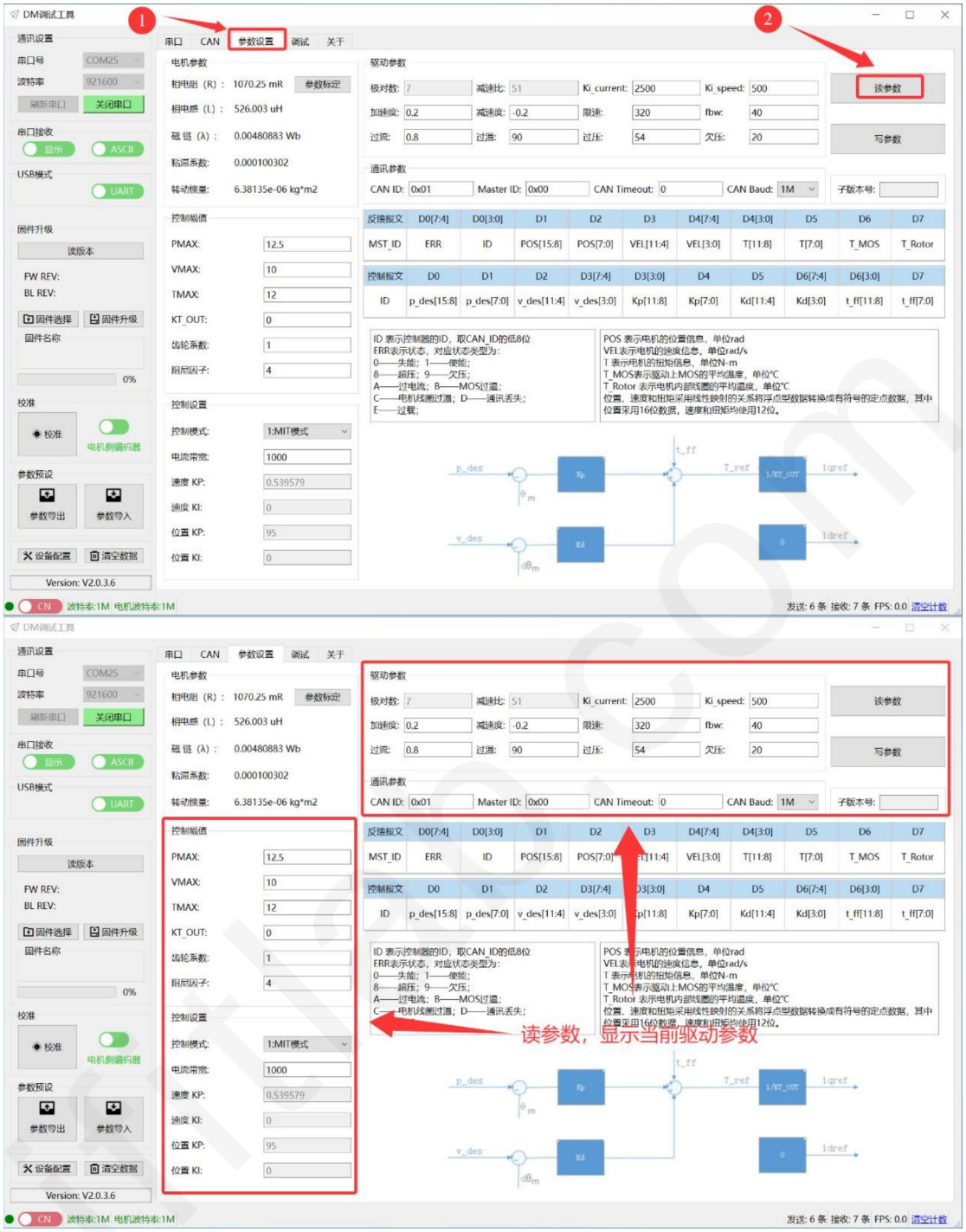
Step 3: Calibrate data validation

After calibration, the drive automatically performs data verification. If the deviation is too large, an error (type 3) will be reported with a red light flashing. This error usually occurs due to an output shaft encoder malfunction. Contact after-sales service for assistance.

Serial port read/write parameters

1. Read parameters

Under the Parameters tab, click the Read Parameters button. The driver will upload the parameters used for storage to the Debug Assistant. Check the parameters carefully.



The read parameters include:

(1) Drive Parameters: The parameter settings section for driving

Pole count: The motor's pole count is automatically determined through calibration and must not be modified.

Under-voltage: If the power supply voltage falls below the set value, the driver cannot control the motor operation. The default value is 20V.

Overvoltage: This parameter defines the maximum operating voltage for the driver. During power-up, the motor's voltage is measured. If the detected voltage exceeds this threshold, the driver triggers an error and deactivates. Persistent overvoltage during startup will result in continuous error alerts.

Acceleration/Deceleration: This parameter is used in non-MIT mode to regulate the drive unit's acceleration/deceleration and motor rotor parameters, measured in Krad/s², where deceleration values are negative.

Reducer ratio: The motor's reducer ratio primarily determines its output speed and position, which indirectly affects torque feedback. Do not modify it.

Overtemperature: The motor coil temperature protection threshold is configurable based on operational requirements, with a recommended maximum of 100℃. When exceeded, the driver will deactivate, enter an inoperative state, and trigger an error.

CAN_ID: The driver ID number used for CAN commands, represented in hexadecimal. It is recommended to set it below 16 to avoid conflicts with error codes.

Master ID: The frame ID used by the driver for feedback, in hexadecimal format;

CAN Timeout: A 32-bit integer setting that defines the duration (in 50μs cycles) after which no CAN command is detected, triggering motor protection. This setting applies only when the motor is enabled.

Speed limit: The maximum rotational speed of the motor rotor (before deceleration), measured in radians per second (rad/s).

Overcurrent: A percentage setting that limits the motor's maximum phase current.

CAN baud rate: Configure the CAN baud rate, supporting 125Kbps to 5Mbps

Ki_current (current loop enhancement factor): an auxiliary control parameter for the current loop. Modification is not recommended.

fbw (speed loop filter bandwidth): the rotational speed filter bandwidth, measured in Hz.

Subversion: The firmware subversion number

Ki_speed (Speed Loop Enhancement Coefficient): A speed loop auxiliary control parameter, with an adjustable range recommended between 200-800. Note: Upon successful parameter reading, the debugging interface parameters (e.g., PMAX, ID) will be updated automatically.

(2) Motor Parameters: Section on Motor Parameters

These parameters are automatically identified by the driver. When the driver board is replaced, the parameters need to be recalibrated, but they are automatically saved in the driver.

(3) Control amplitude: Set the parameter range of the drive command

MAX: In MIT mode, it is the mapped value received as commands by the motor; in other modes, it serves as feedback information for the motor.

Mapping values. For mapping rules, see the CAN Communication section.

VMAX: Same as PMAX;

TMAX: Same as PMAX;

KT_OUT: The torque coefficient of the motor, set to 0 when the motor parameters are accurately identified.

Gear coefficient: The gear torque transmission coefficient, with a value ≤ 1.0.

damping factor: the bandwidth ratio between current loop and speed loop control, not utilized.

(4) Control Settings

Control modes: MIT mode, position-speed mode, speed mode, force-position hybrid mode, period-synchronized position mode, period-synchronized speed mode, and period-synchronized torque mode.

current bandwidth: Set the gain coefficient of the current loop, default is 1000.

speed KP/KI and position KP/KI: parameters for the speed loop and position loop.

After writing the parameters, read them and update the host computer to ensure the feedback data is correct.

2. Write parameters

Check the drive parameters, control amplitude, and control settings. Modify the parameters as needed, then click 'Write Parameters' to save the changes to the drive.

Clicking the 'Write Parameters' button will reset the chip automatically. To prevent safety hazards, ensure the motor is in the 'Disengaged' state when using this function.

[Note]：

1 Do not modify the logarithm of pole and reduction ratio parameters.

2 After clicking "Write Parameters", the driver will automatically restart without requiring external power.

3 After completing the parameter writing, read the parameters and update them to the host computer to ensure the feedback data is accurate.

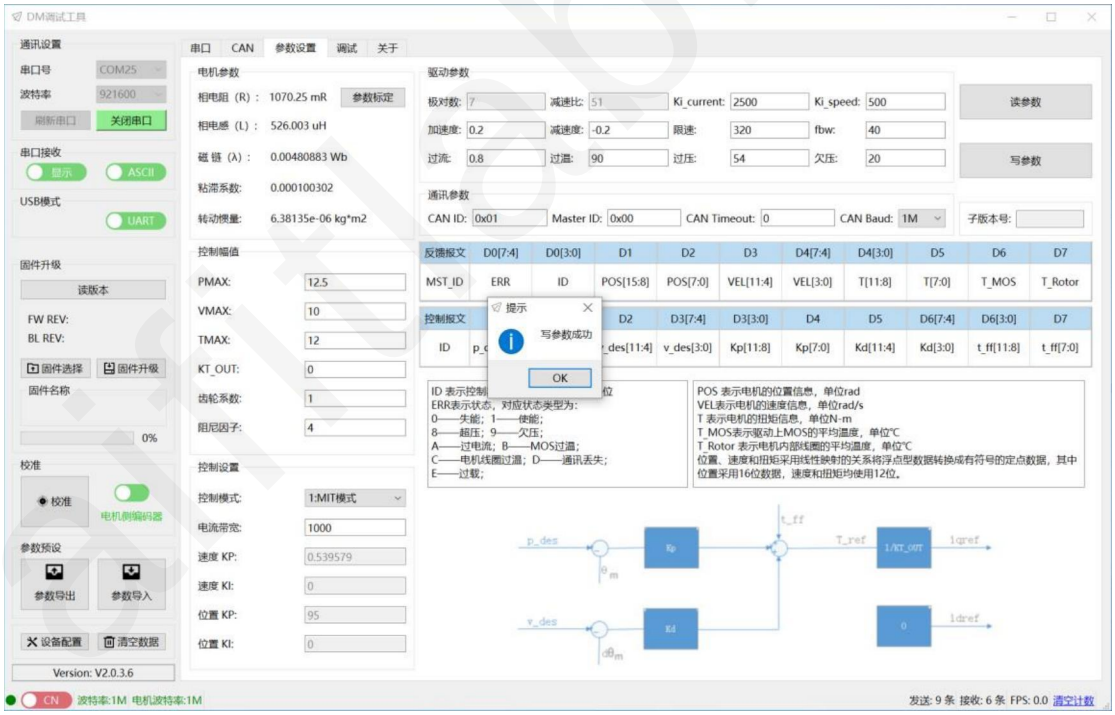
Shakedown test

This function is available only when a CAN interface is connected. It can debug a single motor only. Connect the CAN cable to the driver board before operation. Before debugging, confirm the wiring sequence and current control mode, then select the appropriate sub-label card in the debugging page based on the mode.

1. Selection and Confirmation of Control Modes

In the Debug Assistant, go to 'Parameter Settings' and select 'Control Mode' under Control Settings. You can choose from seven modes: MIT mode, speed-position mode, speed mode, force-position hybrid control, period-synchronized position mode, period-synchronized speed mode, and period-synchronized torque mode. Then click 'Write Parameters' to finalize the control mode settings.

When you change the control mode, a prompt window will appear: "Parameter written successfully."



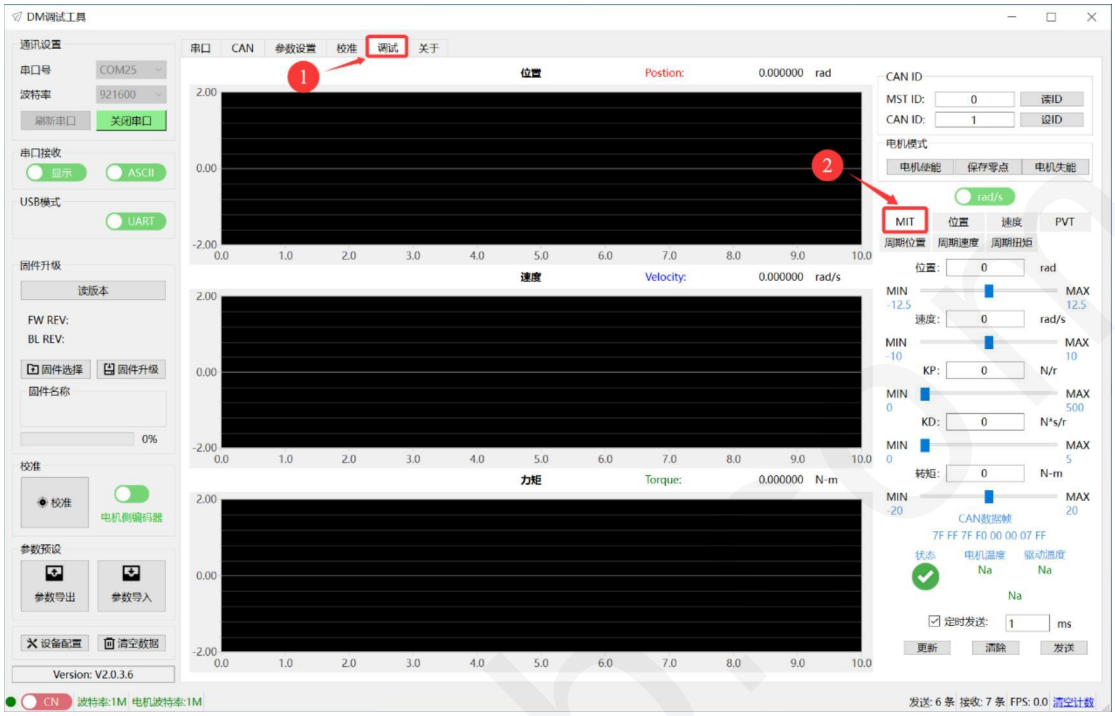
(1) The control mode is determined by analyzing the serial port print data from the power supply; the mode indicated by the arrow represents the current active control mode.

(2) Alternatively, the drive's control mode can be determined by checking the displayed information after re-reading parameters on the parameter setting page.

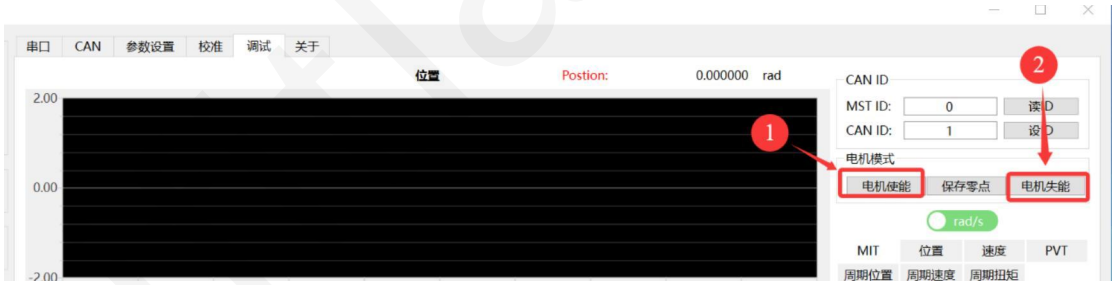
After writing the parameters, read them and update the host computer to ensure the feedback data is correct.

2.MIT pattern

(1) Reference: Select and confirm the control mode. Set the current mode to MIT, then choose the corresponding MIT sub-tab in the debugging interface.



(2) Verify the CAN ID is correct (available through serial port printing, parameter settings, or on the debug page via read/write buttons).



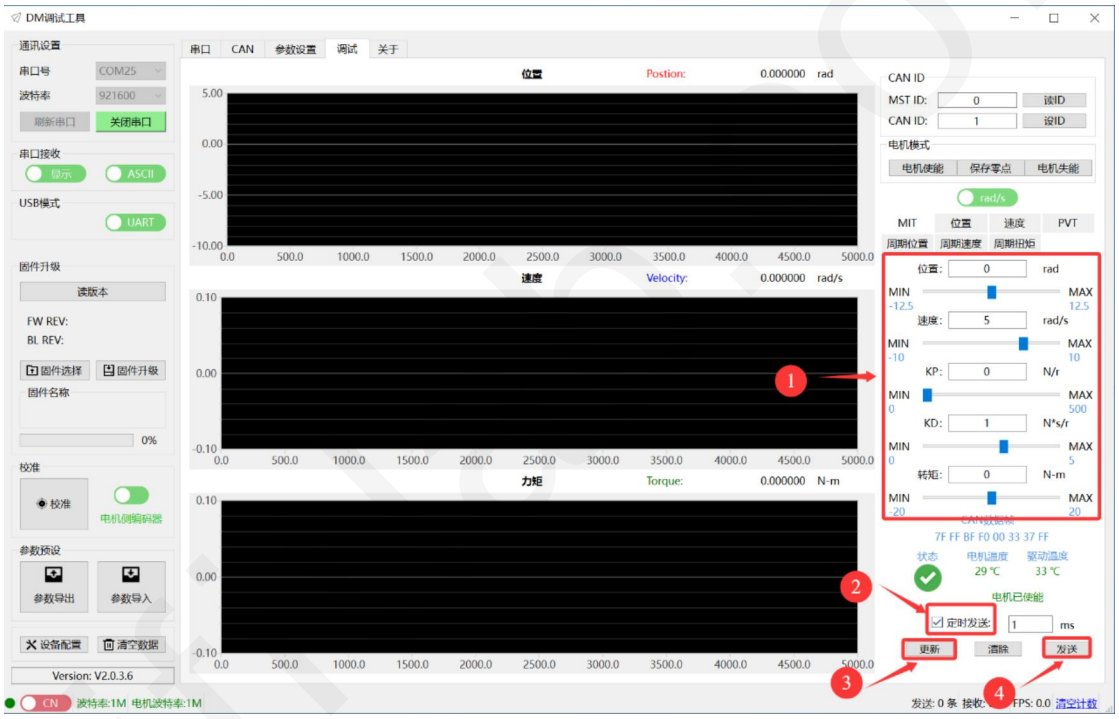
(3) The MIT mode features three control methods: speed, position, and torque.

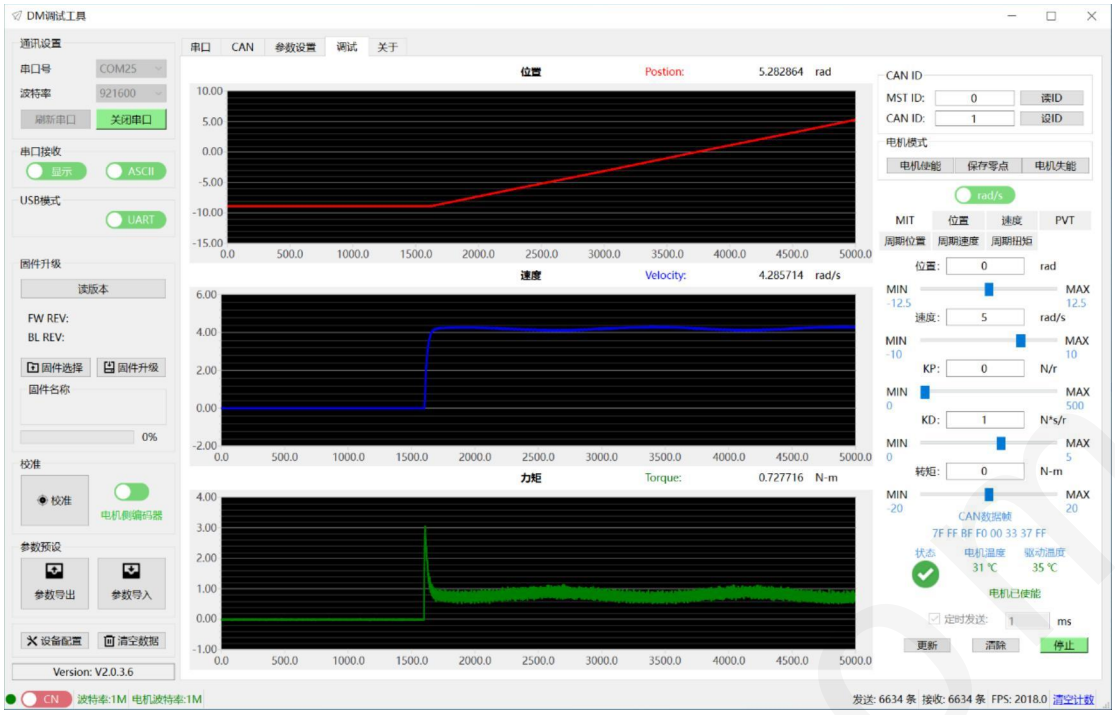
① speed control

Step 1: Click the Enable button in the motor mode bar. The driver's green light will turn on, indicating the motor is enabled.

Step 2: Set the motor to operate at the specified speed.

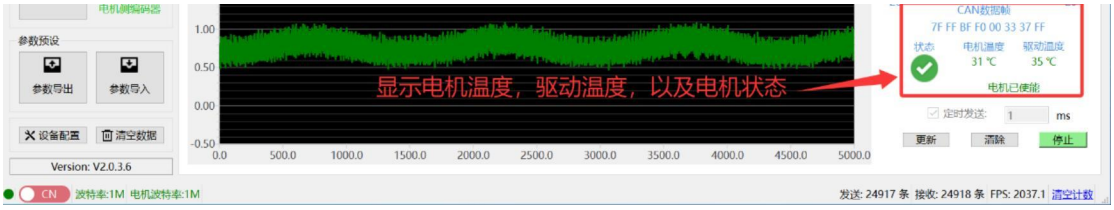
For example: Set the speed to 5 rad/s, KD to 1 N·s/r, and all other values to 0. Check the 'Scheduled Send' box, then click the 'Update' and 'Send' buttons in sequence. The parameter curve changes can be viewed in the debugging interface. Note: Keep the motor stationary.



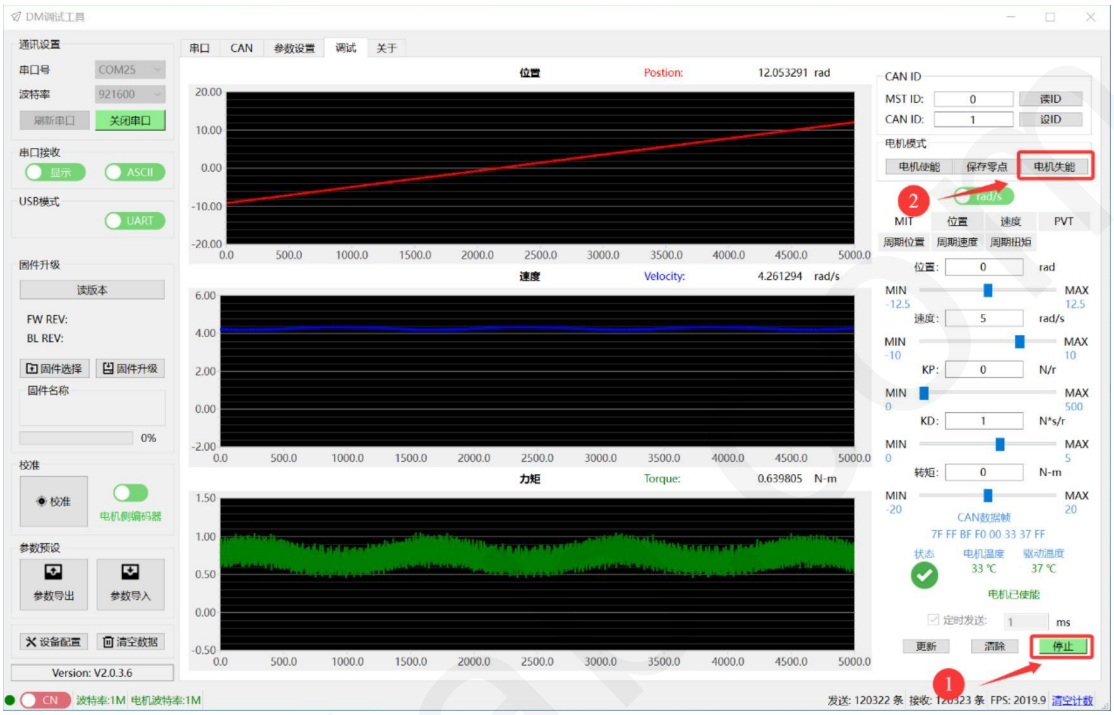


To view debugging changes, modify control parameters in the original interface. Keep the 'Scheduled Send' option checked and click the 'Update' button to proceed with debugging.

The debugging assistant interface displays real-time temperature data and operational status of the motor and drive. The feedback frame can also be viewed for this information. For details on the feedback frame format and status types, see the "4.1 Feedback Frame" section.

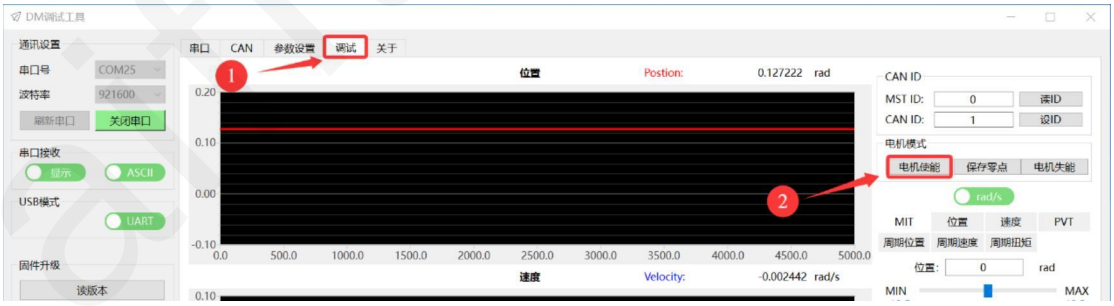


Step 3: To exit debug mode, click the "Stop" and "Disable" buttons in sequence. The red light on the drive will turn on, indicating the motor mode has been exited.



② position control

Step 1: Click the Enable button in the motor mode bar. The driver's green light will turn on, indicating the motor is enabled.

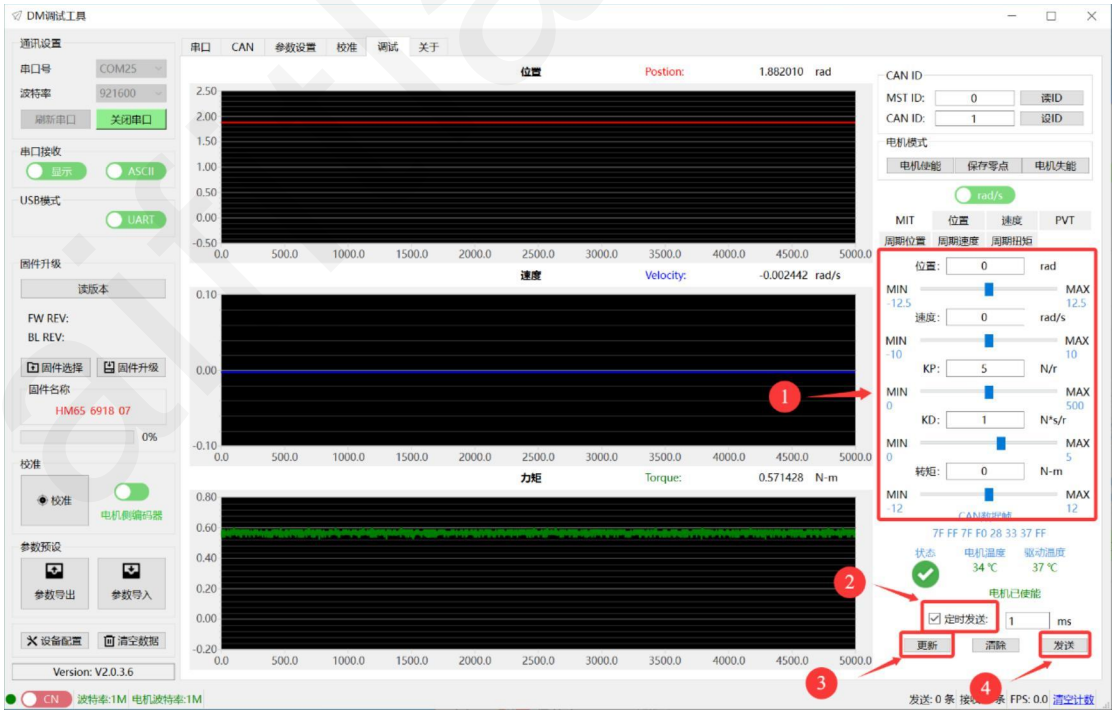


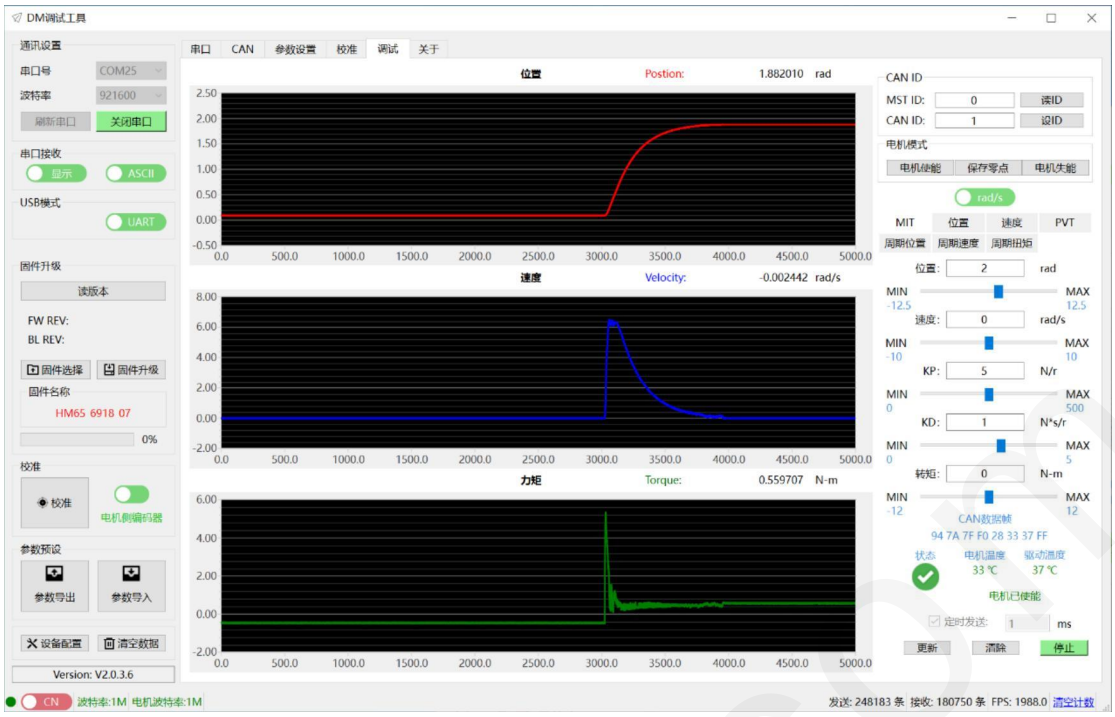
Step 2: Rotate the motor to the specified position.

When setting the 'position' parameter in the control parameters, ensure it closely matches the motor's initial position to prevent excessive deviation that could cause motor shock.

To set the motor's current position as zero, click' Save Zero Point' in the control command bar for easy position parameter configuration.

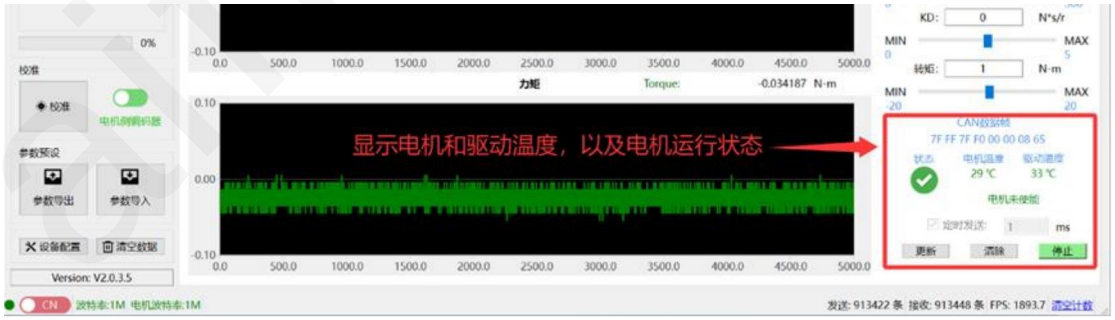
For example, with the following parameters: position set to 2 rad, KP set to 5 N/r, KD set to 1 N·s/r, and all other parameters set to 0, check the 'Scheduled Send' box, then click the 'Update' and 'Send' buttons in sequence. The parameter curve changes can be viewed in the debugging interface. Note: Keep the motor stationary.



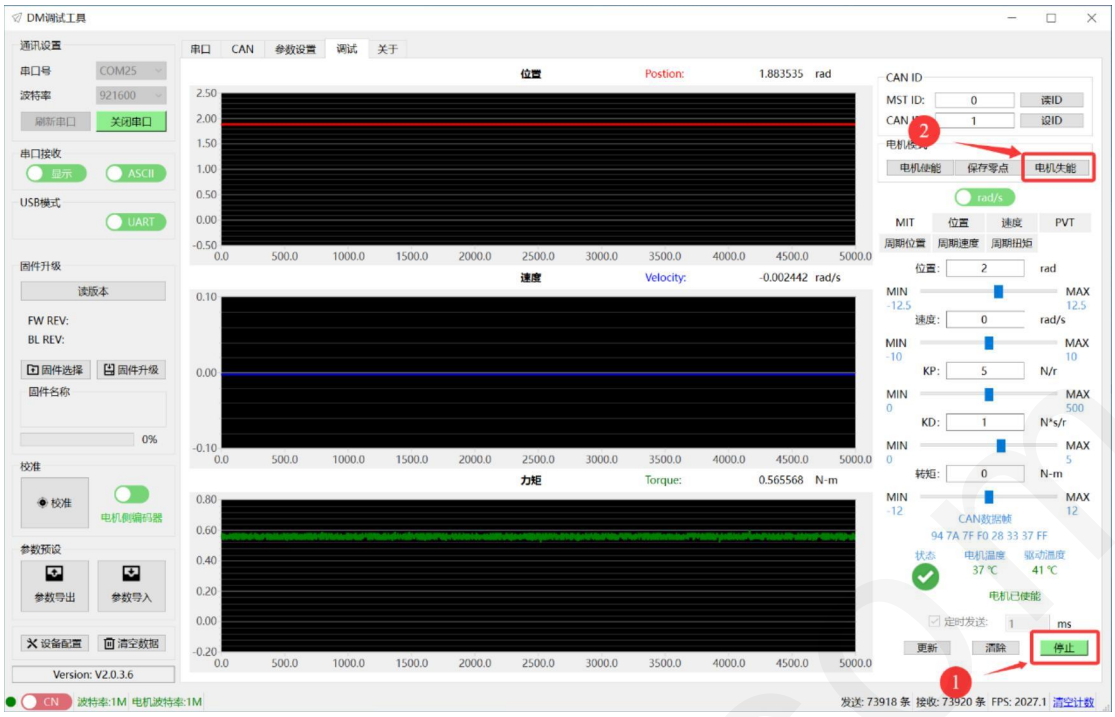


To view debugging changes, modify control parameters in the original interface. Keep the 'Scheduled Send' option checked and click the 'Update' button to proceed with debugging.

The debugging assistant interface displays real-time temperature data and operational status of the motor and drive. The feedback frame can also be viewed for this information. For details on the feedback frame format and status types, see the "4.1 Feedback Frame" section.



Step 3: To exit debug mode, first stop the motor, then sequentially press the "Stop" and "Disable" buttons. The driver's red light will illuminate, indicating the motor mode has been exited.



③ MIT torque control

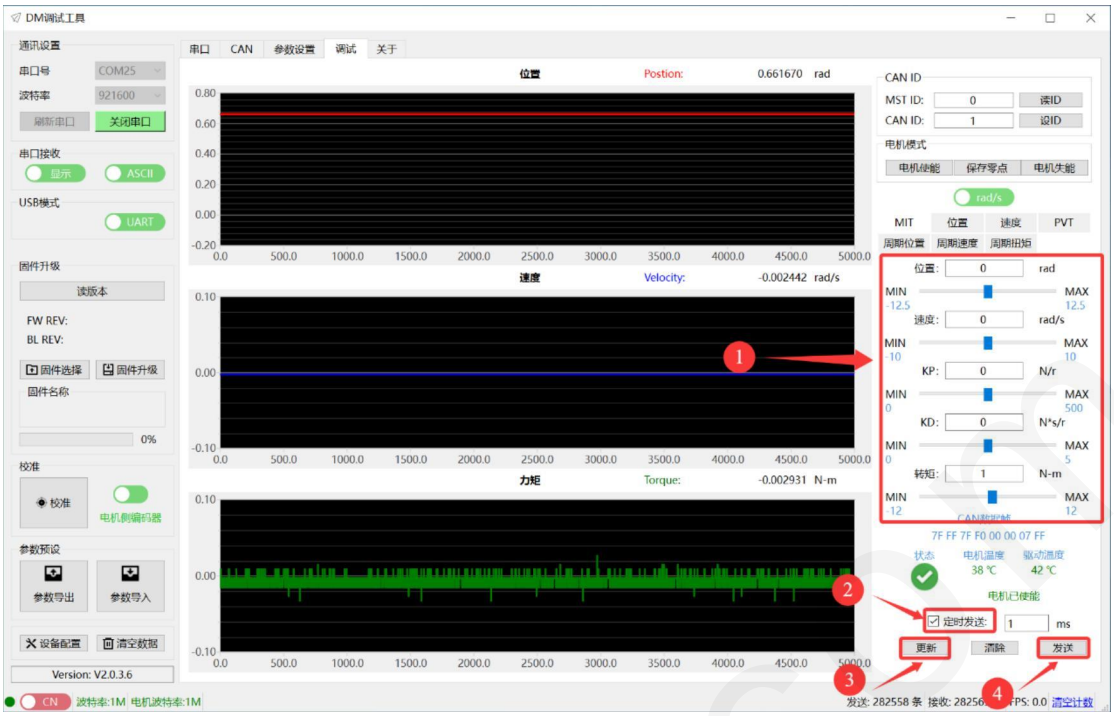
Step 1: Click the Enable button in the motor mode bar. The driver's green light will turn on, indicating the motor is enabled.

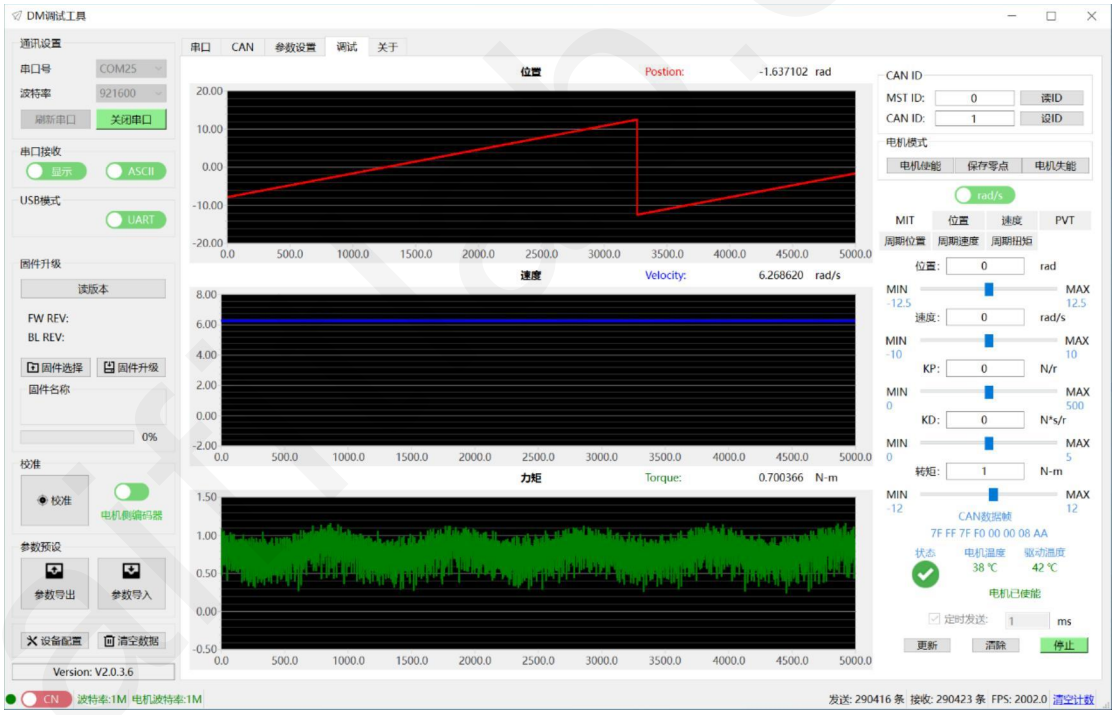
The second step is to debug the motor with a given torque.

Note: When unloaded, the motor will accelerate to maximum speed even with a minimal torque.

For example, set the torque to 1N-m with all other parameters at 0, check the Scheduled Send' box, then click 'Update' and 'Send' buttons in sequence. The parameter curve changes will appear in the debug interface. Note: Keep the motor

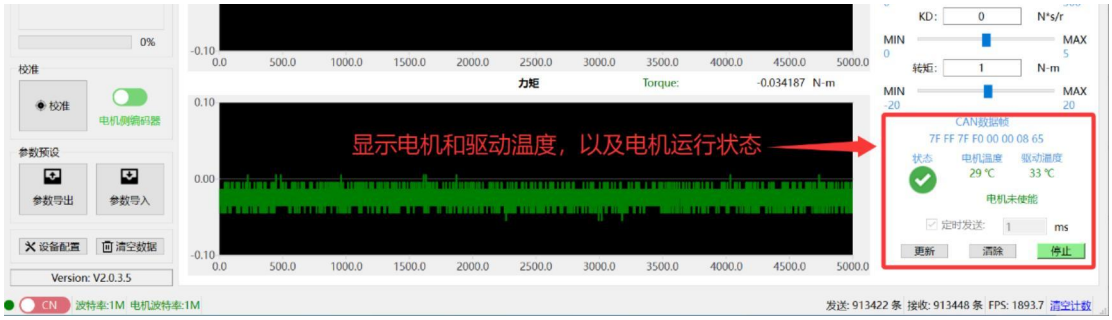
stationary.



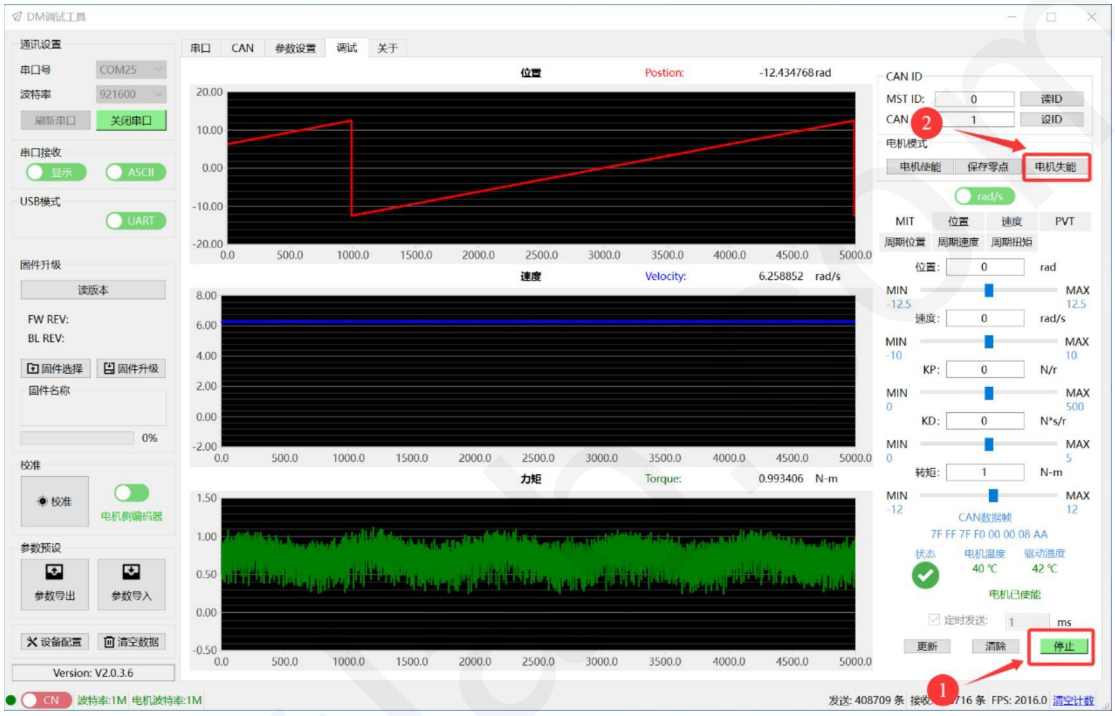


To debug, modify the control parameters to view changes. Edit parameters directly in the original interface, keep the 'Scheduled Send' option checked, and click the 'Update' button to proceed.

The debugging assistant interface displays real-time temperature data and operational status of the motor and drive. The feedback frame can also be viewed for this information. For details on the feedback frame format and status types, see the "4.1 Feedback Frame" section.

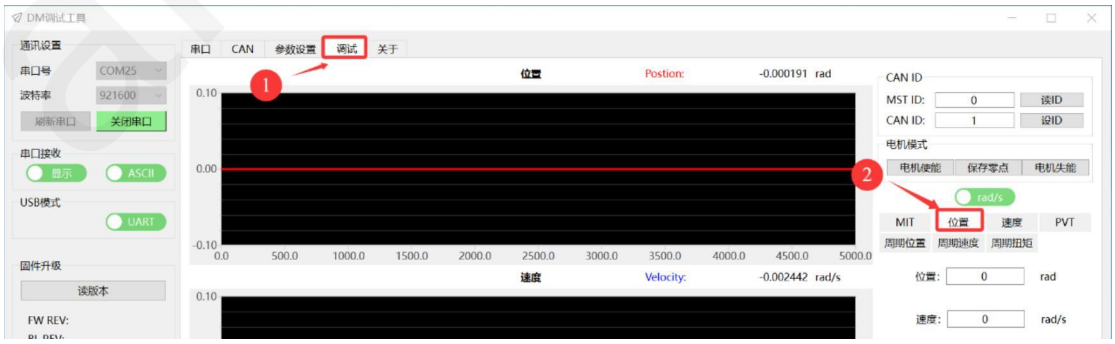


Step 3: To exit debug mode, click the "Stop" and "Enable" buttons in sequence. The red light on the drive will turn on, indicating the motor mode has been exited.

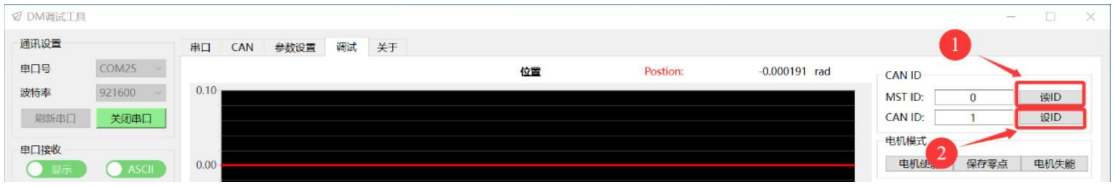


3. Position Velocity Mode

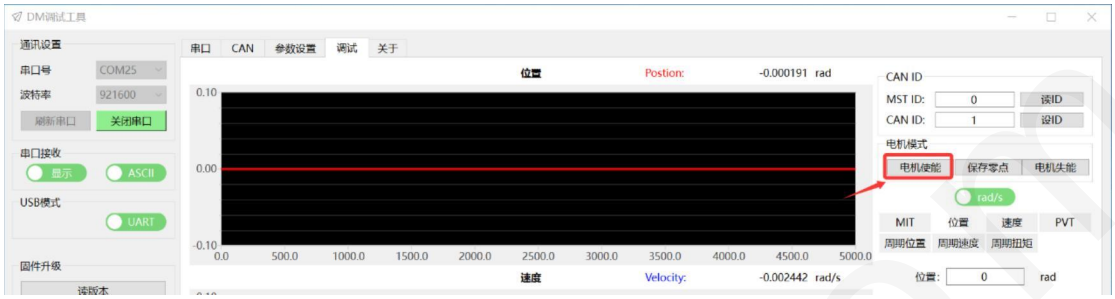
(1) To debug the position-speed mode, switch the motor mode to position-speed mode on the parameter page, then click 'Write Parameters' to apply the changes. Select the corresponding 'Position' sub-tab in the debugging interface.



(2) Verify the CAN ID is correct (available through serial port printing, parameter settings, or debugging interface settings)



(3) Click the "Enable" button in the motor mode section. The driver's green light will illuminate, indicating the motor is now enabled.



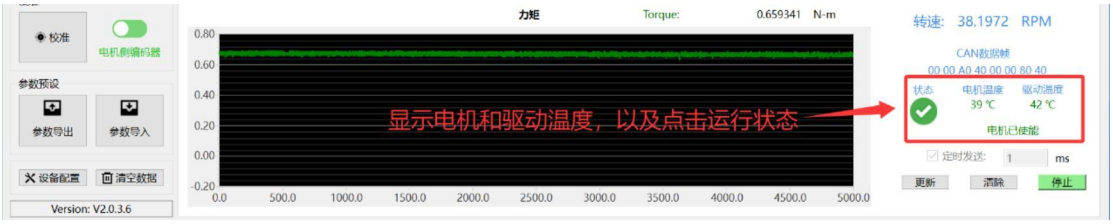
(4) Set the parameters, then the motor will run at the preset speed to the designated position.

Before setting parameters, note the motor's initial position to use as a reference.

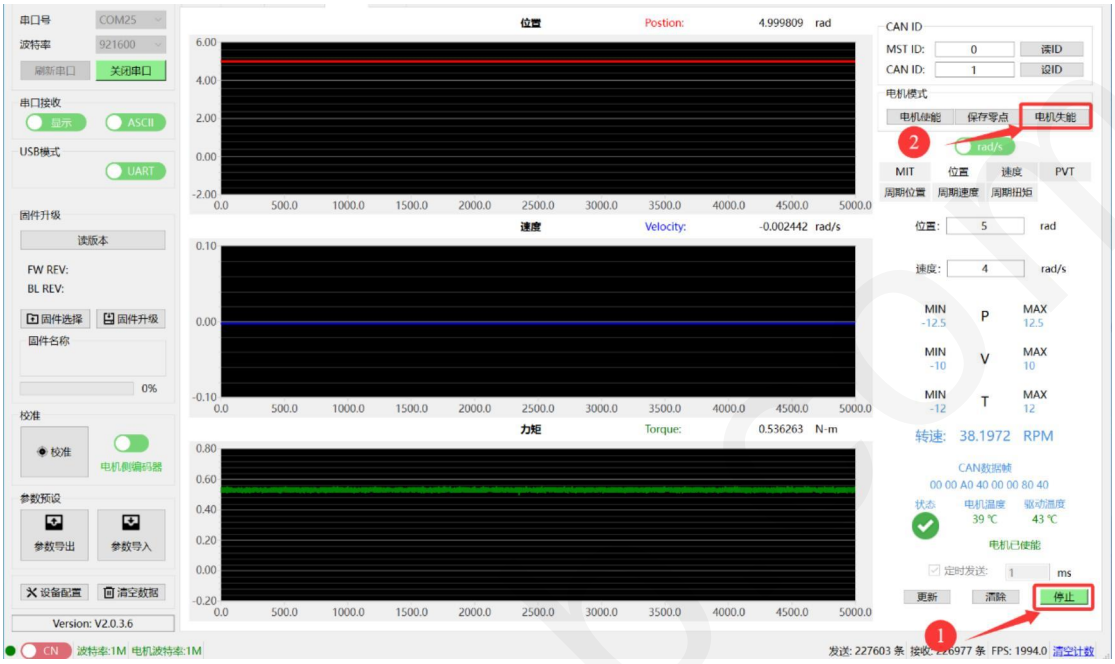
For example, set the parameters: position at 5 rad, speed at 4 rad/s. Check the 'Scheduled Send' box, then click the 'Update' and 'Send' buttons in sequence to view the parameter curve changes in the debugging interface. Note: Keep the motor stationary.

To view debugging changes, modify control parameters in the original interface. Keep the 'Scheduled Send' option checked and click the 'Update' button to proceed with debugging.

The debugging assistant interface displays real-time temperature data and operational status of the motor and drive. The feedback frame can also be viewed for this information. For details on the feedback frame format and status types, see the "4.1 Feedback Frame" section.

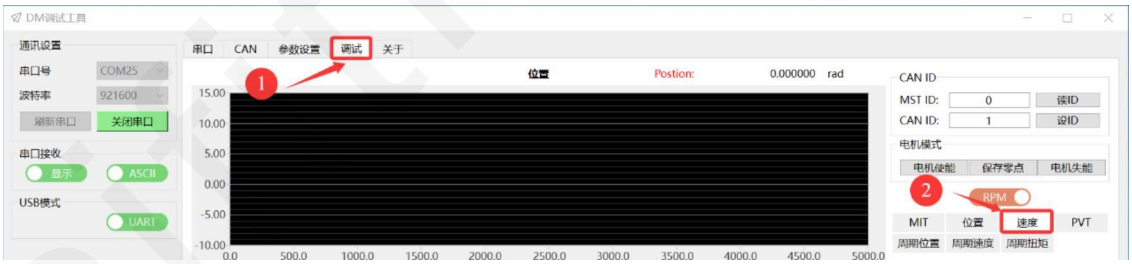


(5) To exit debug mode, press the "Stop" and "Disable" buttons in sequence. The driver's red light will illuminate, indicating the motor mode has been exited.



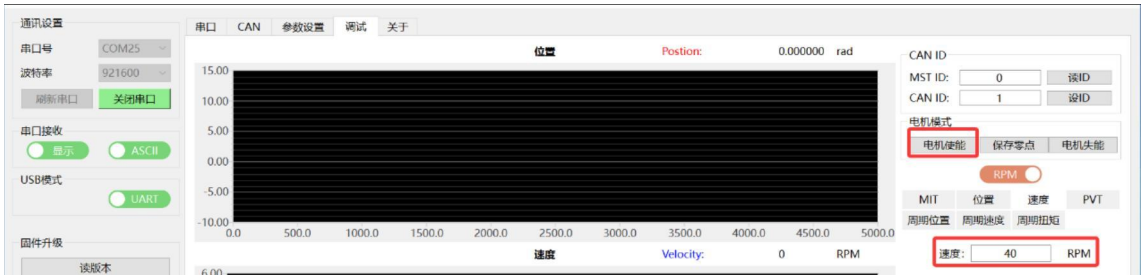
4. Speed Mode

(1) Select the speed mode as the current control mode by confirming the reference control mode, then choose the corresponding 'Speed' sub-tab card on the debugging page.



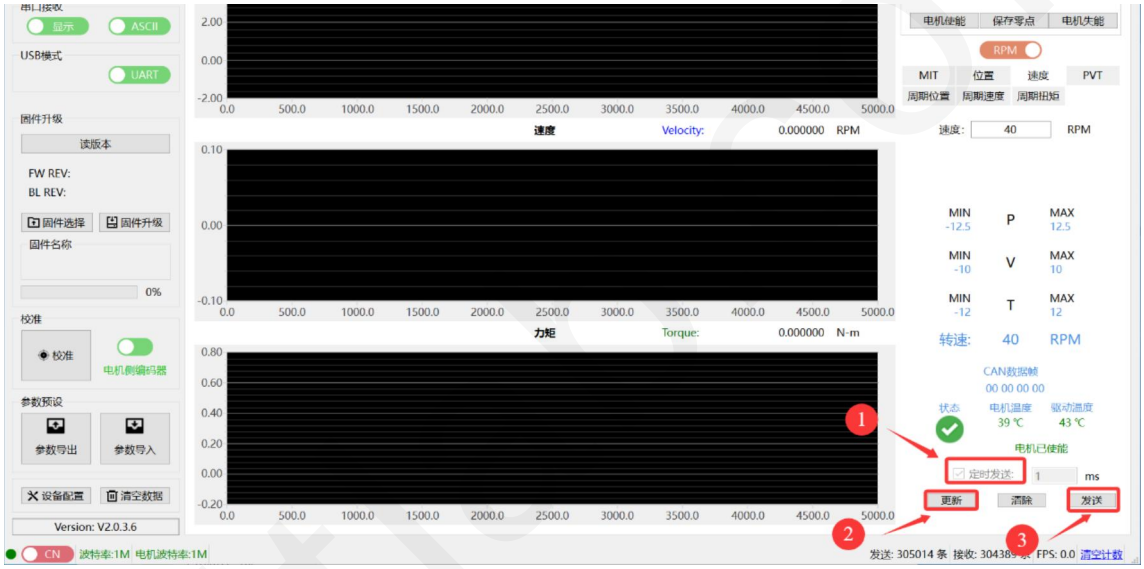
(2) Verify the CAN ID is correct (available through serial port printing, parameter settings, or debugging interface settings)

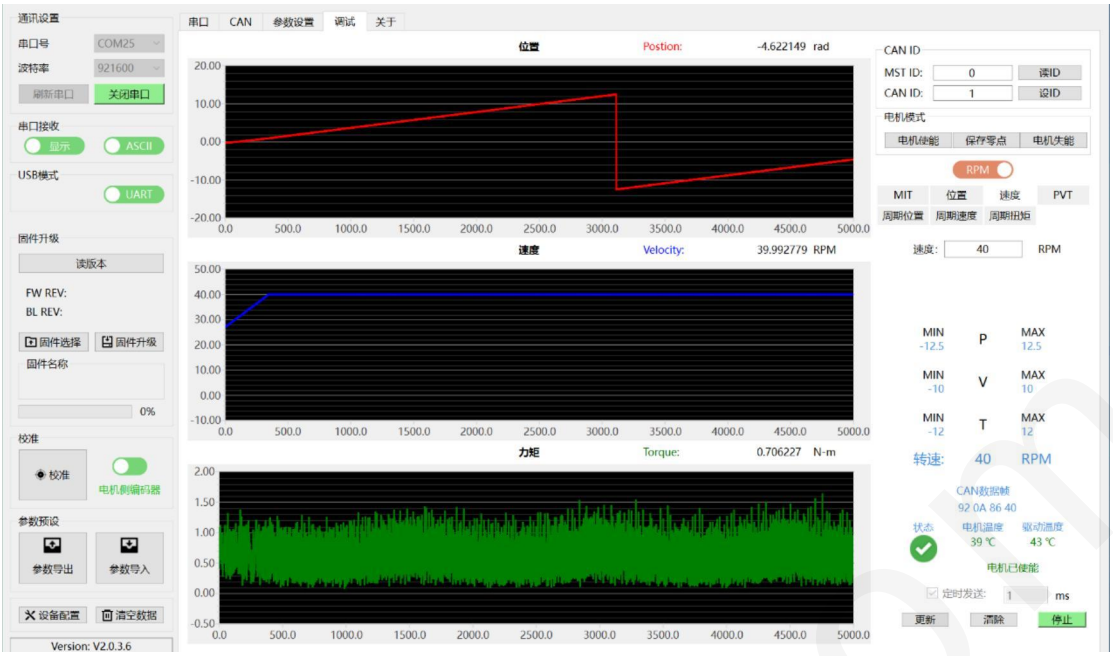
(3) Click the "Enable" button in the motor mode section. The driver's green light will illuminate, indicating the motor is now enabled. For example, set the speed to 40 rpm.



(4) Set the parameters, and the motor will operate at the preset speed.

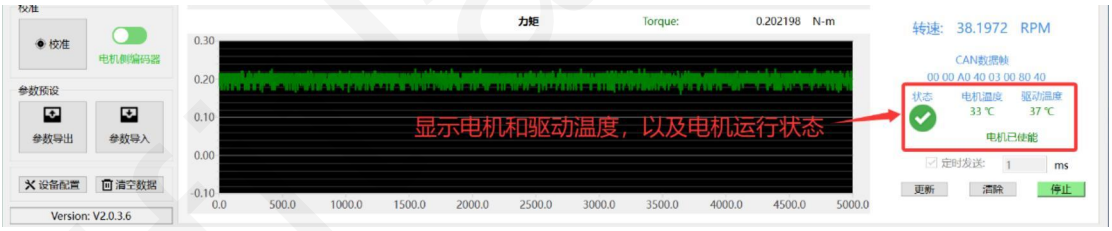
Check the 'Scheduled Send' box, then click the 'Update' and 'Send' buttons in sequence to view the parameter curve changes in the debugging interface. Note: Keep the motor stationary.



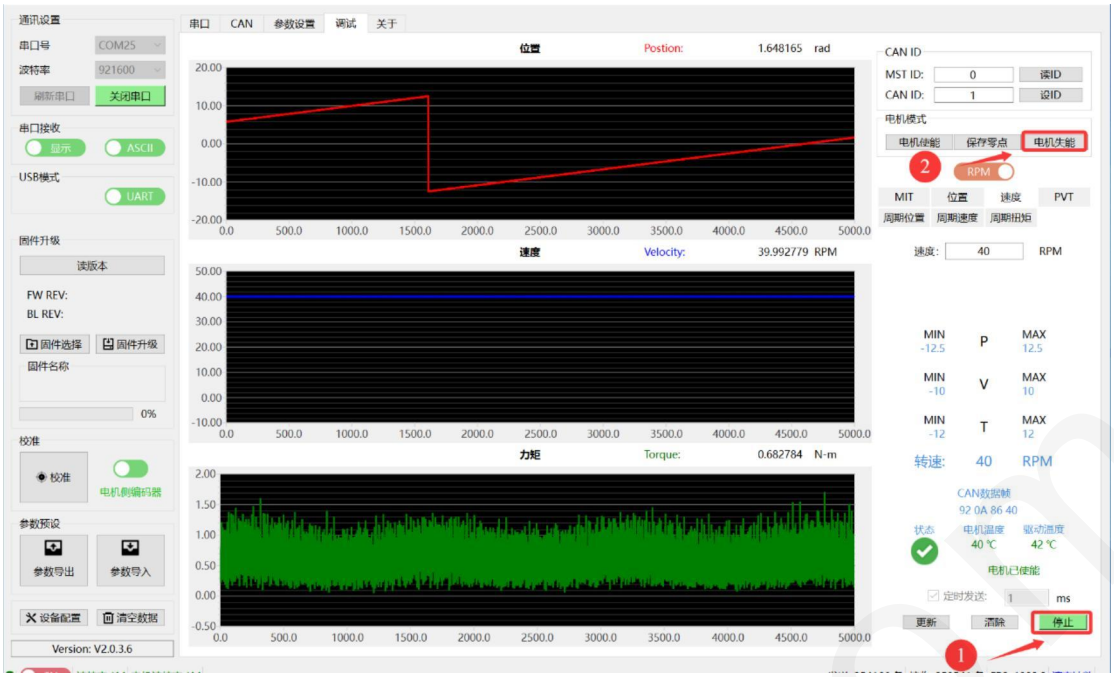


To view debugging changes, modify control parameters in the original interface. Keep the 'Scheduled Send' option checked and click the 'Update' button to proceed with debugging.

The debugging assistant interface displays real-time temperature data and operational status of the motor and drive. The feedback frame can also be viewed for this information. For details on the feedback frame format and status types, see the "4.1 Feedback Frame" section.

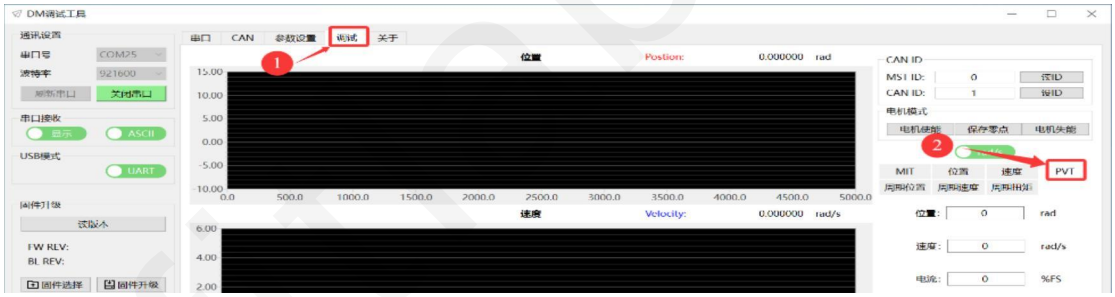


(5) To exit debug mode, first set the speed to 0 to stop the motor, then sequentially press the "Stop" and "Disable" buttons. The driver's red light will illuminate, indicating the motor mode has been exited.



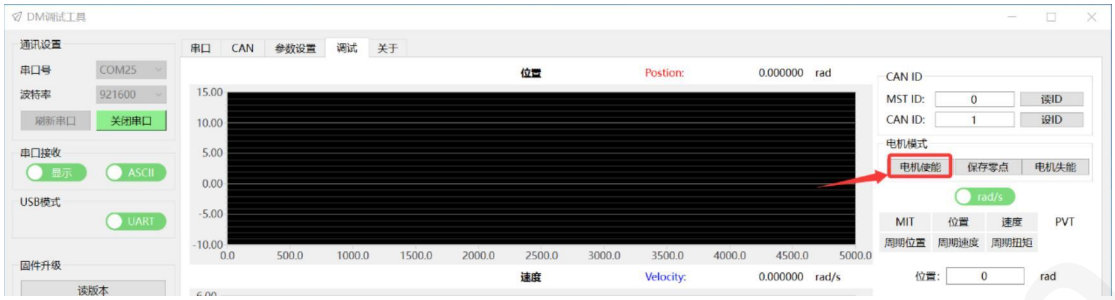
5. Force-position hybrid control mode

(1) To debug the force-position hybrid control mode, switch the motor mode to PVT mode on the parameter page, then click 'Write Parameters' to apply the changes. Select the corresponding 'PVT' sub-tab card in the debugging interface.



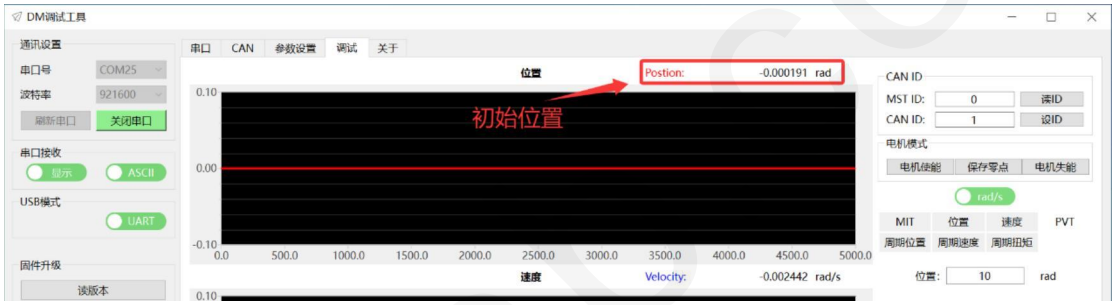
(2) Verify the CAN ID is correct (available through serial port printing, parameter settings, or debugging interface settings)

(3) Click the "Enable" button in the motor mode section. The driver's green light will illuminate, indicating the motor is now enabled.

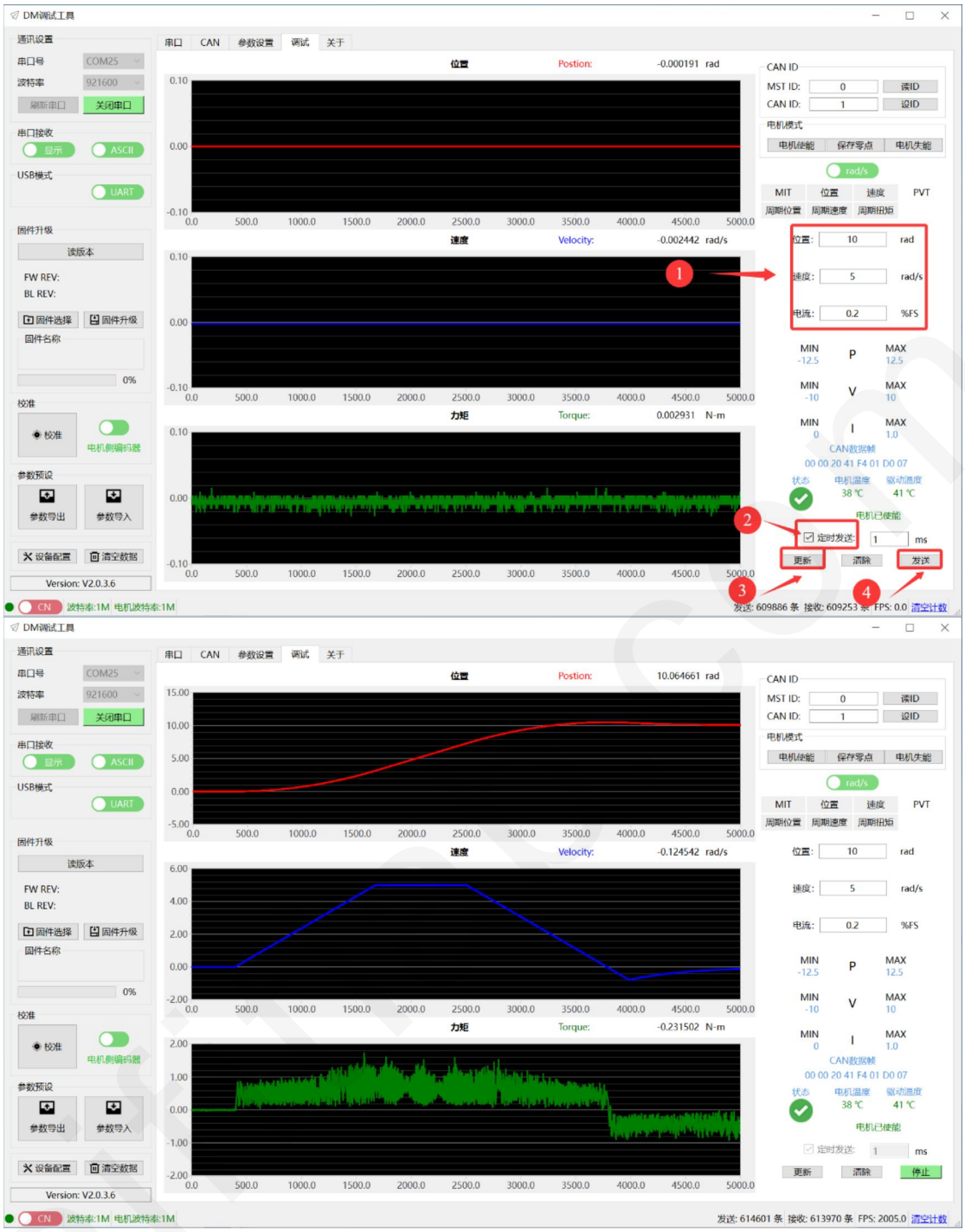


(4) Set the parameters, then the motor will run at the preset speed to the designated position.

Before setting parameters, note the motor's initial position to use as a reference.

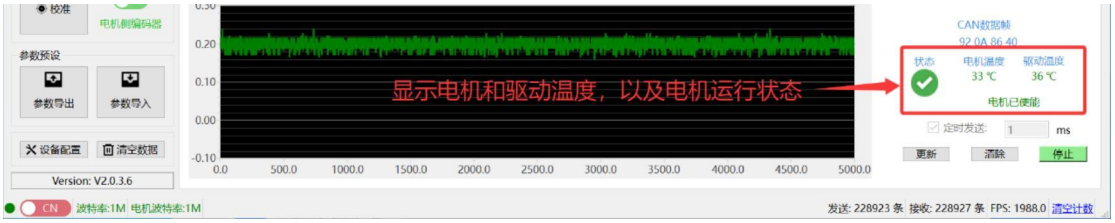


For example, set the parameters: position 10 rad, speed 5 rad/s, and current 20%. Check the 'Scheduled Send' box, then click the 'Update' and 'Send' buttons in sequence to view the parameter curve changes in the debugging interface. Note: Keep the motor stationary.

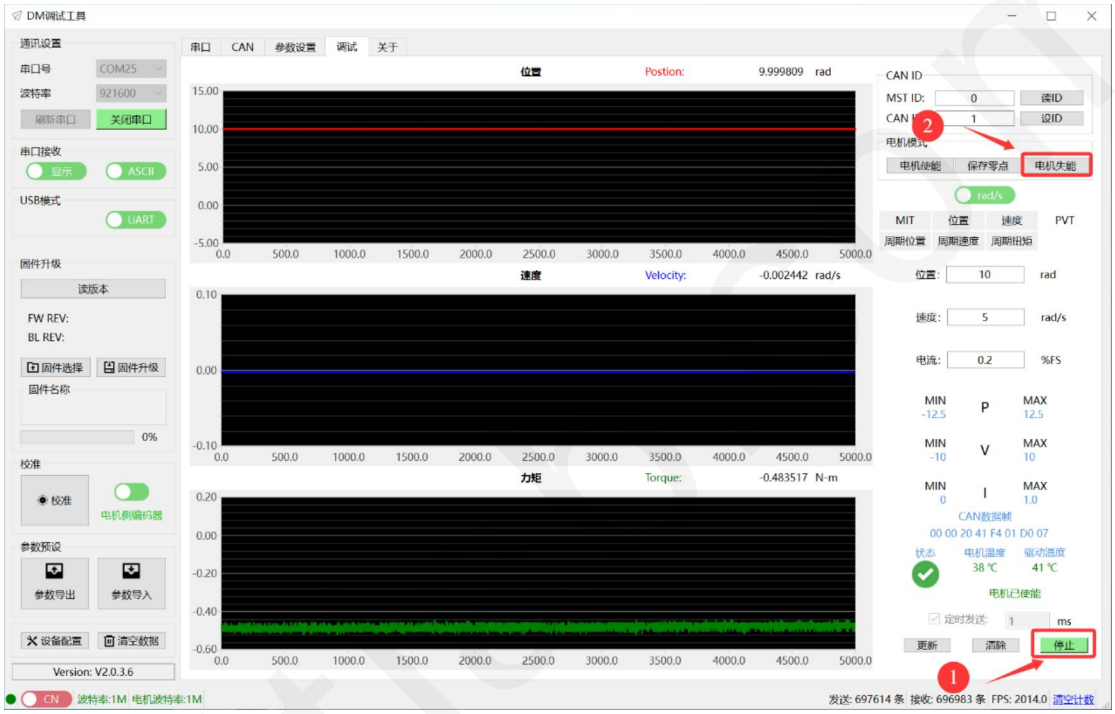


To view debugging changes, modify control parameters in the original interface. Keep the 'Scheduled Send' option checked and click the 'Update' button to proceed with debugging.

The debugging assistant interface displays real-time temperature data and operational status of the motor and drive. The feedback frame can also be viewed for this information. For details on the feedback frame format and status types, see the "4.1 Feedback Frame" section.

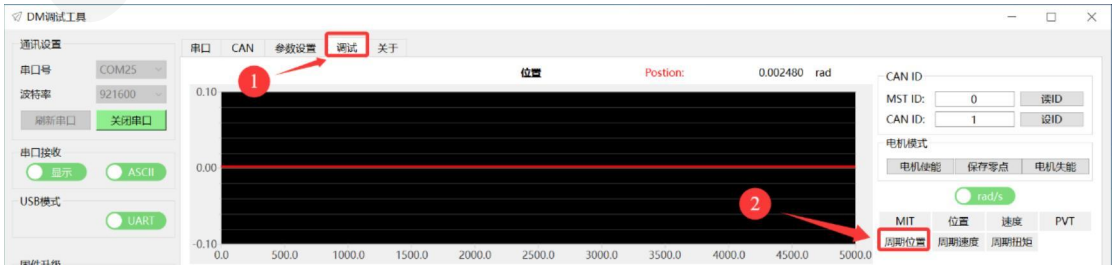


(5) To exit debug mode, first ensure the motor has stopped, then sequentially press the "Stop" and "Disable" buttons. The driver's red light will illuminate, indicating the motor mode has been exited.

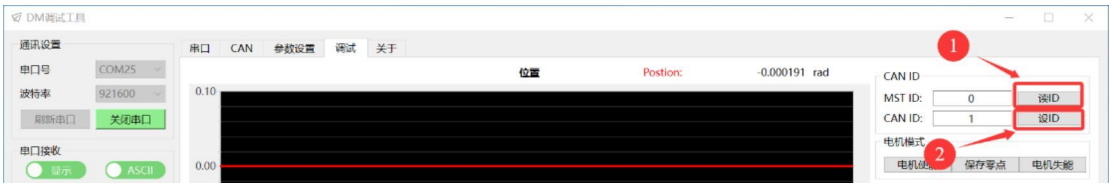


6. Periodic Synchronization Position Mode

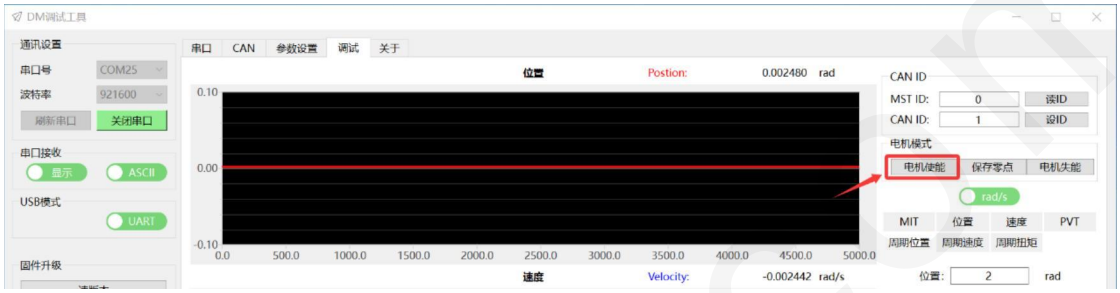
(1) To enable the periodic position synchronization mode, switch the motor mode to this mode on the parameter page, then click 'Write Parameters' to apply the changes. Select the corresponding 'Periodic Position' sub-tab in the debugging interface.



(2) Verify the CAN ID is correct (available through serial port printing, parameter settings, or debugging interface settings)

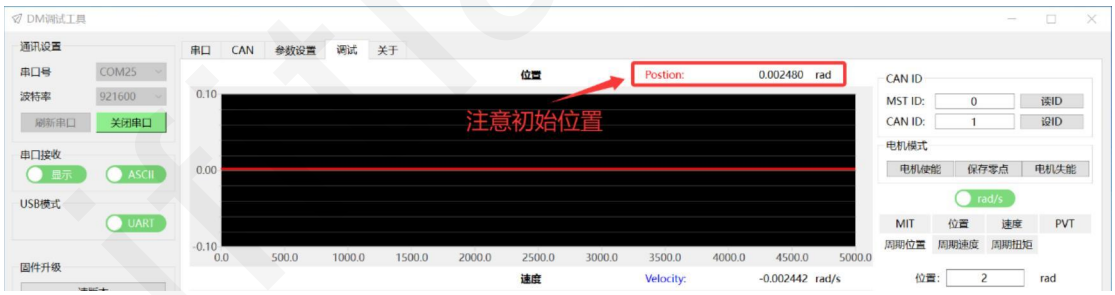


(3) Click the "Enable" button in the motor mode section. The driver's green light will illuminate, indicating the motor is now enabled.



(4) Set the parameters, then the motor will run at the preset speed to the designated position.

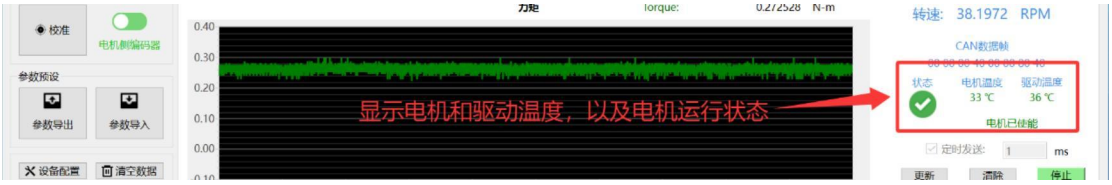
Before setting parameters, note the motor's initial position to use as a reference.



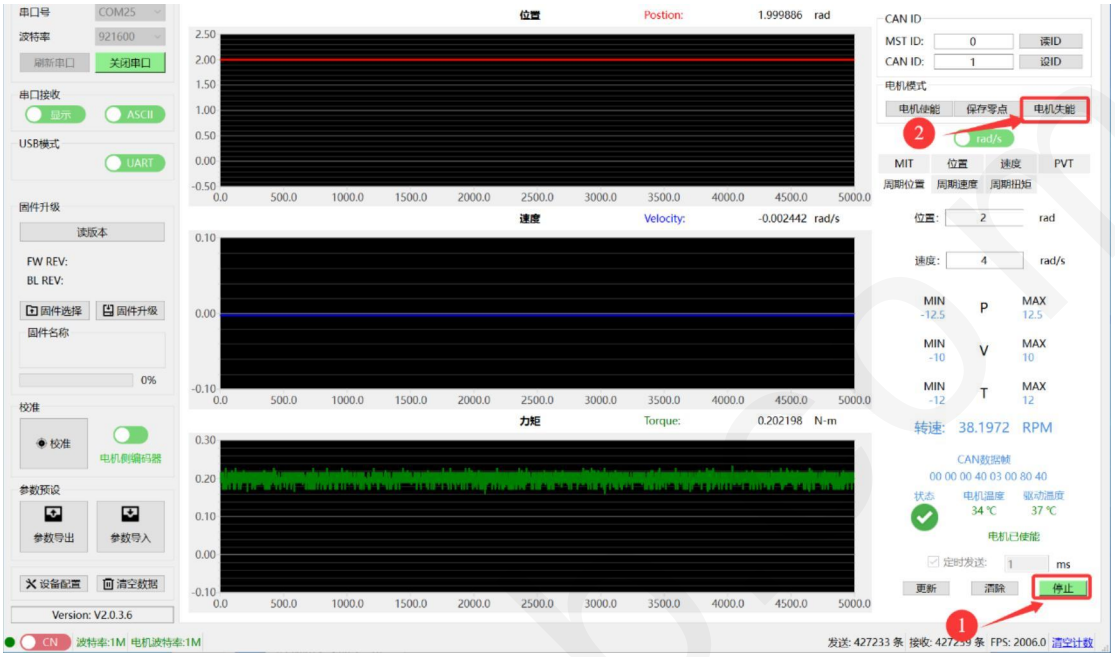
For example, set the parameters: position at 2 rad, speed at 4 rad/s. Check the 'Scheduled Send' box, then click the 'Update' and 'Send' buttons in sequence to view the parameter curve changes in the debugging interface. Note: Keep the motor stationary.

To view debugging changes, modify control parameters in the original interface. Keep the 'Scheduled Send' option checked and click the 'Update' button to proceed with debugging.

The debugging assistant interface displays real-time temperature data and operational status of the motor and drive. The feedback frame can also be viewed for this information. For details on the feedback frame format and status types, see the "4.1 Feedback Frame" section.

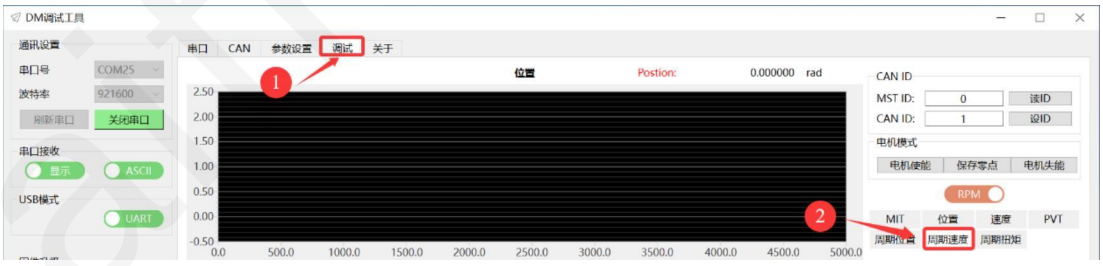


(5) To exit debug mode, press the "Stop" and "Disable" buttons in sequence. The driver's red light will illuminate, indicating the motor mode has been exited.

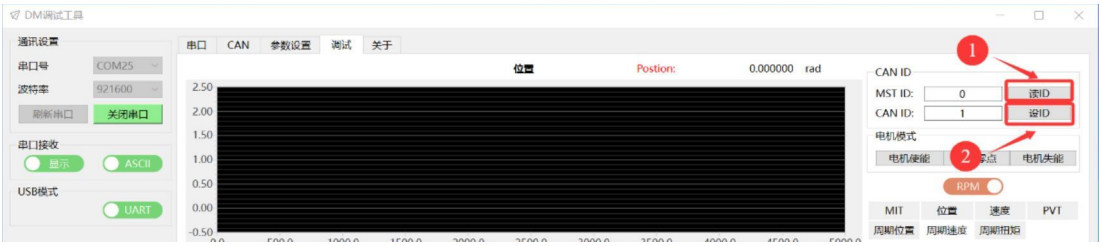


7. Periodic Synchronization Speed Mode

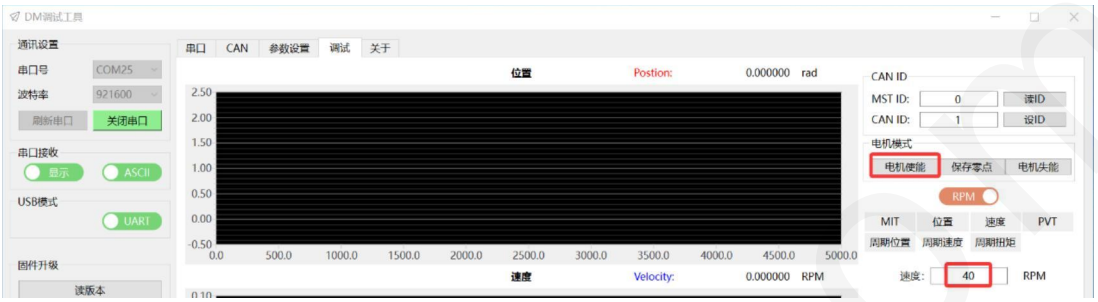
(1) Select the current cycle synchronization speed mode by confirming the reference control mode, then choose the corresponding "Cycle Speed" sub-tab in the debugging interface.



(2) Verify the CAN ID is correct (available through serial port printing, parameter settings, or debugging interface settings)



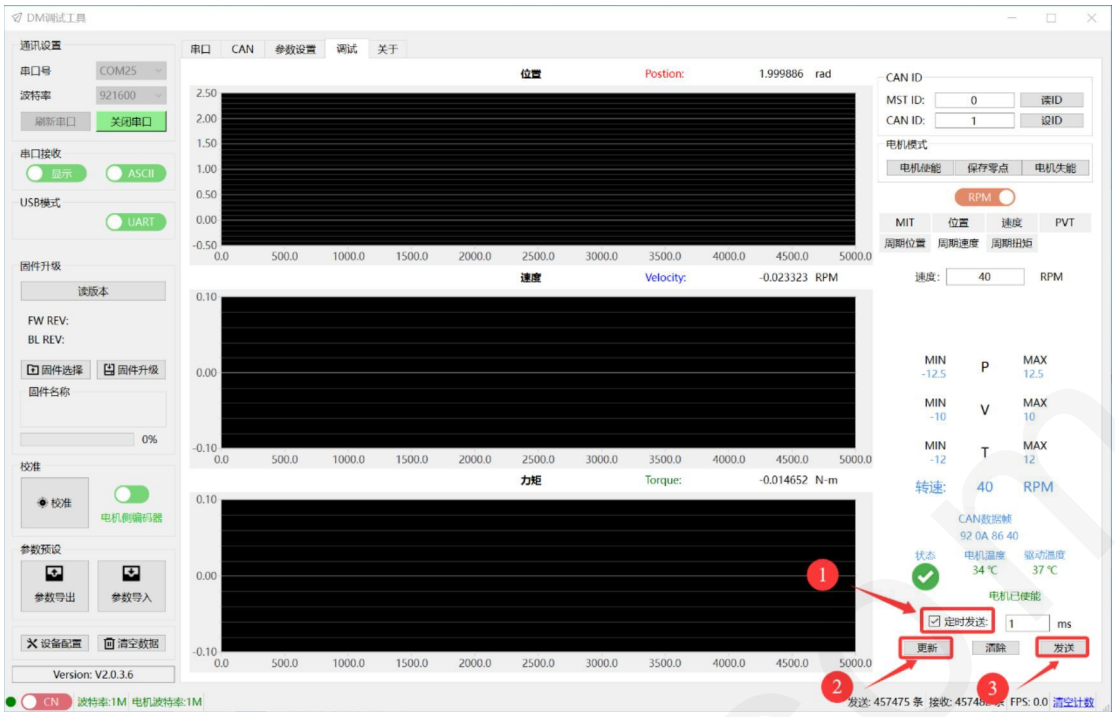
(3) Click the "Enable" button in the motor mode section. The driver's green light will illuminate, indicating the motor is now enabled. For example, set the speed to 40 rpm.

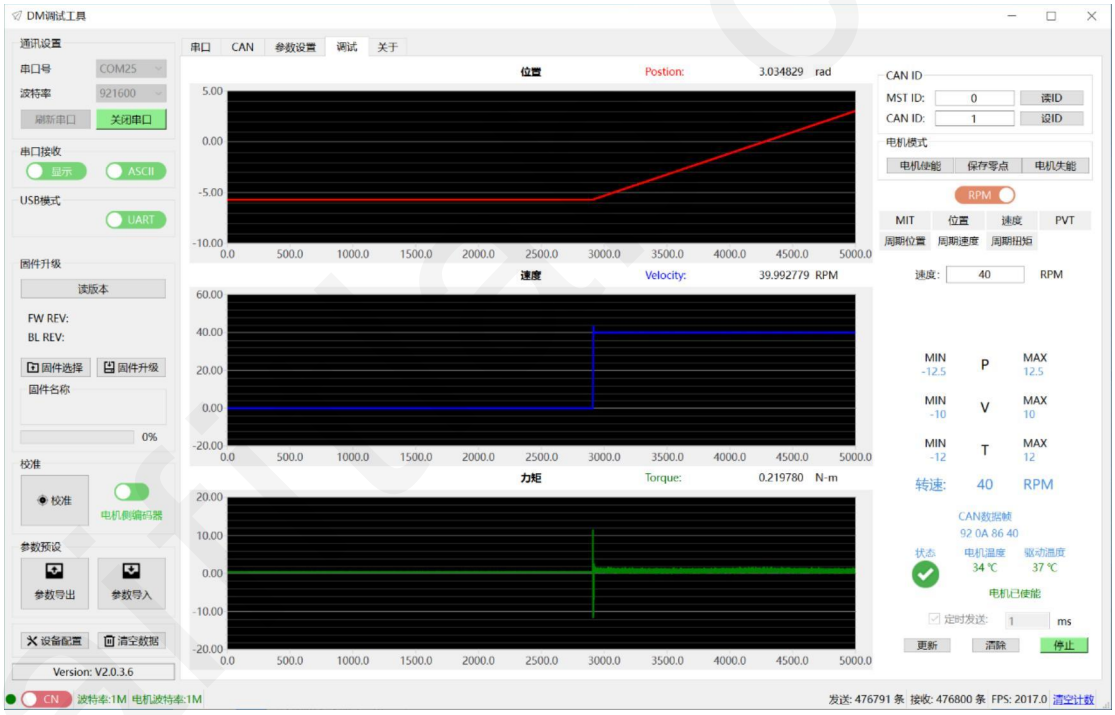


(4) Set the parameters, and the motor will operate at the preset speed.

Check the "Send at scheduled time" box, then click the "Update" and "Send" buttons in sequence to view the parameters in the debugging interface.

Curve change chart. Note to fix the motor.

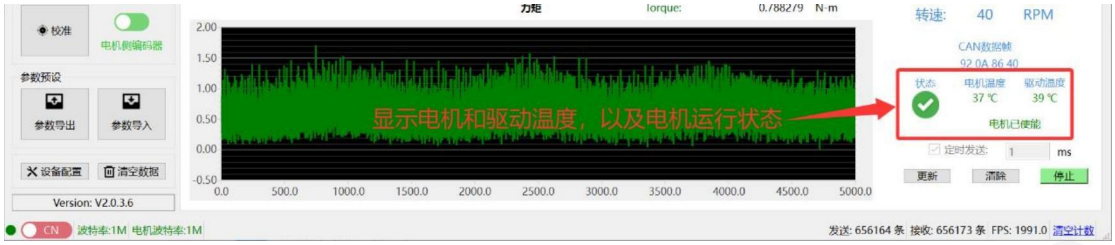




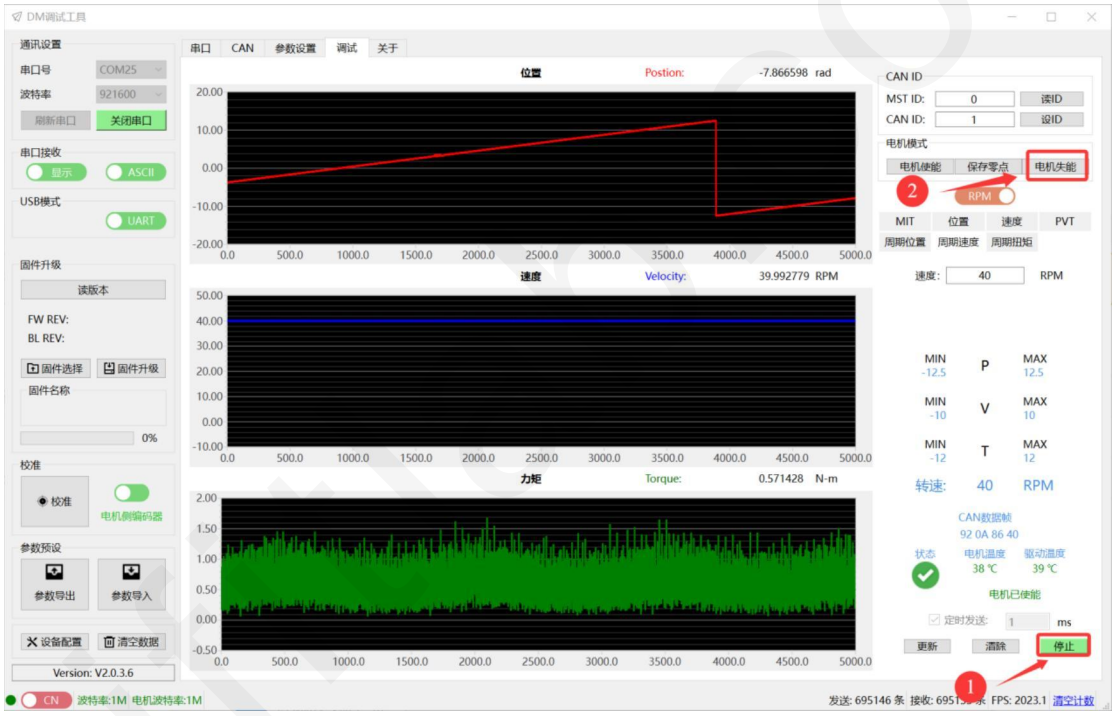
To view debugging changes, modify control parameters in the original interface. Keep the 'Scheduled Send' option checked and click the 'Update' button to proceed with debugging.

The debugging assistant interface displays real-time temperature data and operational status of the motor and drive. The feedback frame can also be viewed for this

information. For details on the feedback frame format and status types, see the "4.1 Feedback Frame" section.



(5) To exit debug mode, first set the speed to 0 to stop the motor, then sequentially press the "Stop" and "Disable" buttons. The driver's red light will illuminate, indicating the motor mode has been exited.



8. Periodic Synchronous Torque Mode

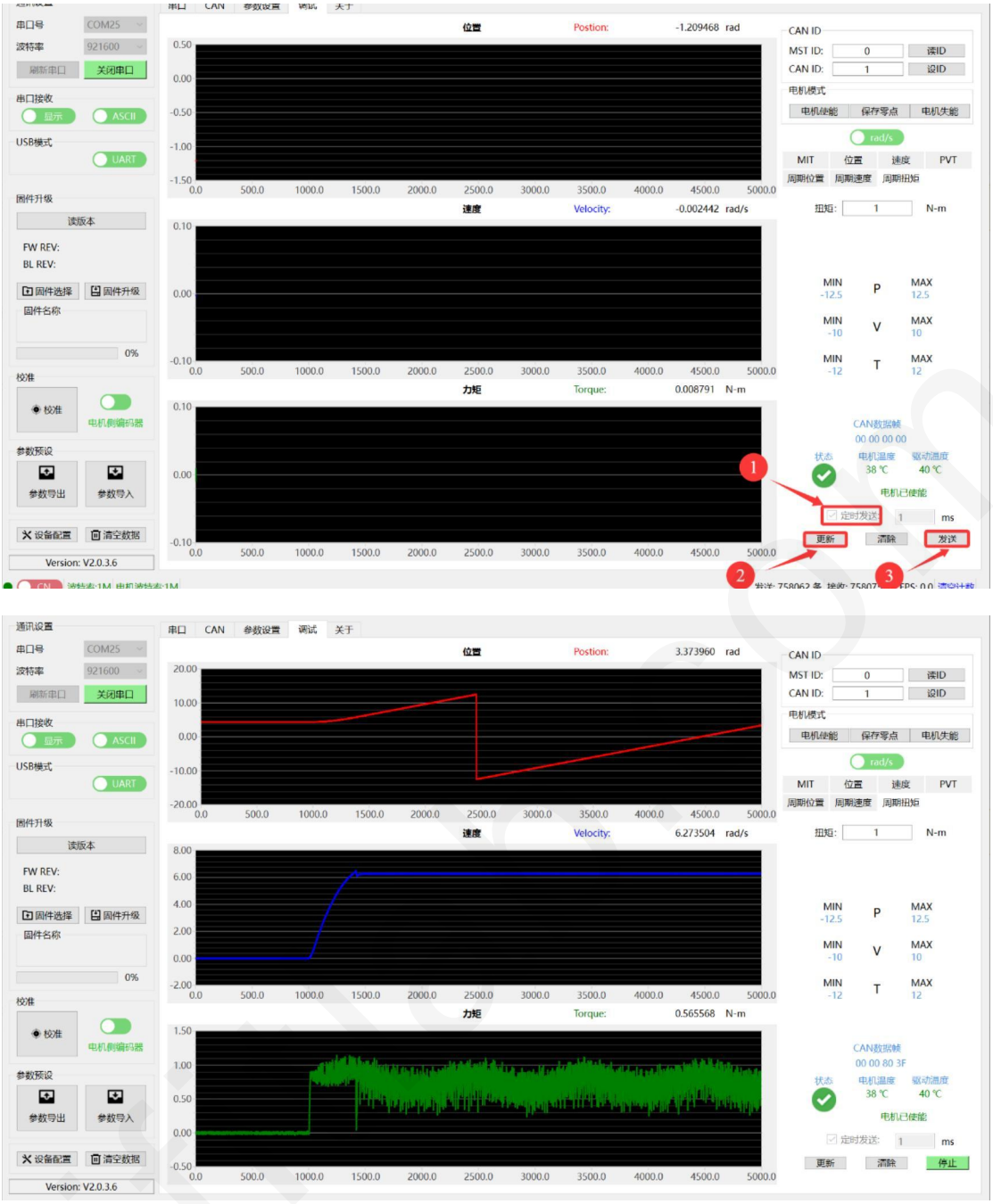
(1) Select the current control mode as the periodic torque synchronization mode, confirm the selection, and choose the corresponding 'Periodic Torque' sub-tab card on the debugging page.

(2) Verify the CAN ID is correct (available through serial port printing, parameter settings, or debugging interface settings)

(3) Click the "Enable" button in the motor mode section. The driver's green light will illuminate, indicating the motor is now enabled. For example, set the torque to 1Nm.

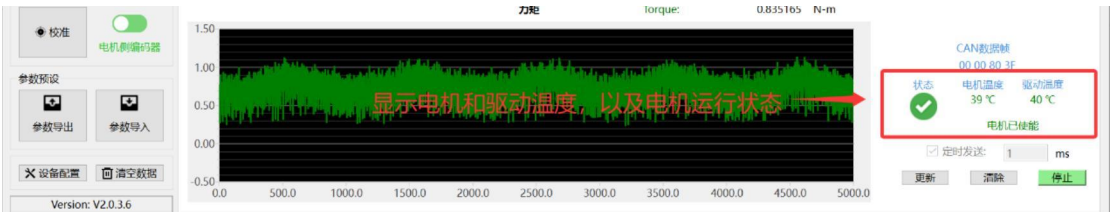
(4) Set the parameters, and the motor will operate at the preset speed.

Check the 'Scheduled Send' box, then click the 'Update' and 'Send' buttons in sequence to view the parameter curve changes in the debugging interface. Note: Keep the motor stationary.

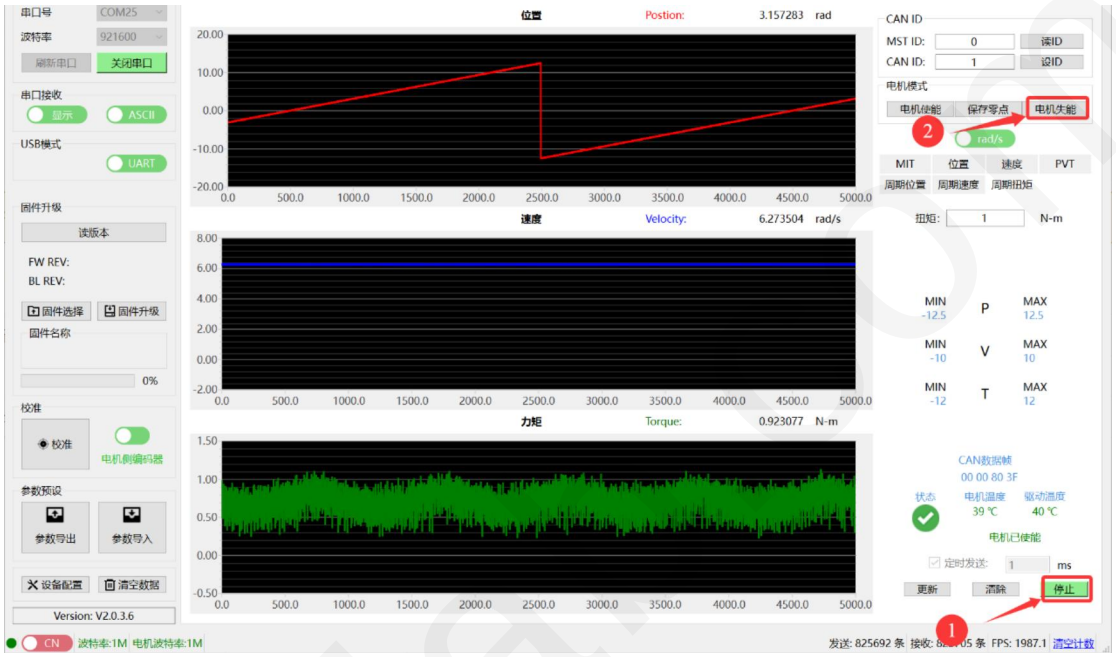


To view debugging changes, modify control parameters in the original interface. Keep the 'Scheduled Send' option checked and click the 'Update' button to proceed with debugging.

The debugging assistant interface displays real-time temperature data and operational status of the motor and drive. The feedback frame can also be viewed for this information. For details on the feedback frame format and status types, see the "4.1 Feedback Frame" section.



(5) To exit debug mode, first apply 0 torque to stop the motor, then sequentially press the "Stop" and "Disable" buttons. The driver's red light will illuminate, indicating the motor mode has been exited.



Firmware Version Check and Upgrade

Version Check

Connect the motor's serial port, CAN port, and power interface. Select the appropriate serial port device and enable it. Click' Read'.

"version", you can view the software version number of the current drive.

Before upgrading the firmware, read the current version. Unless otherwise specified, select firmware with a version number starting with 52 for upgrading. Otherwise, unknown issues may occur, causing unnecessary trouble.

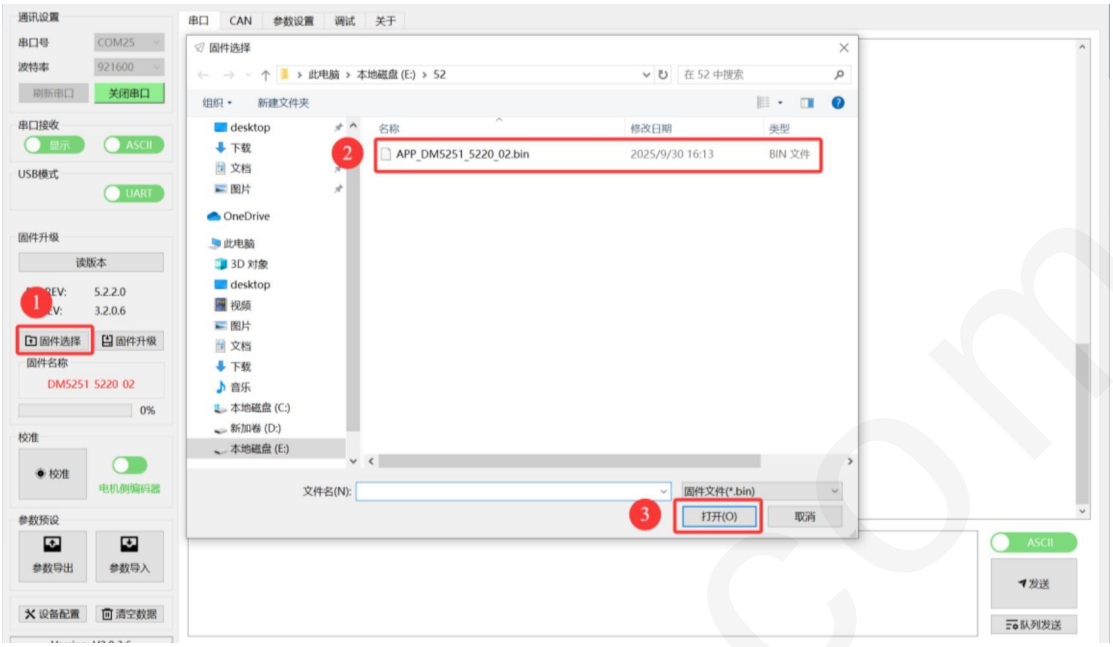
Serial firmware upgrade

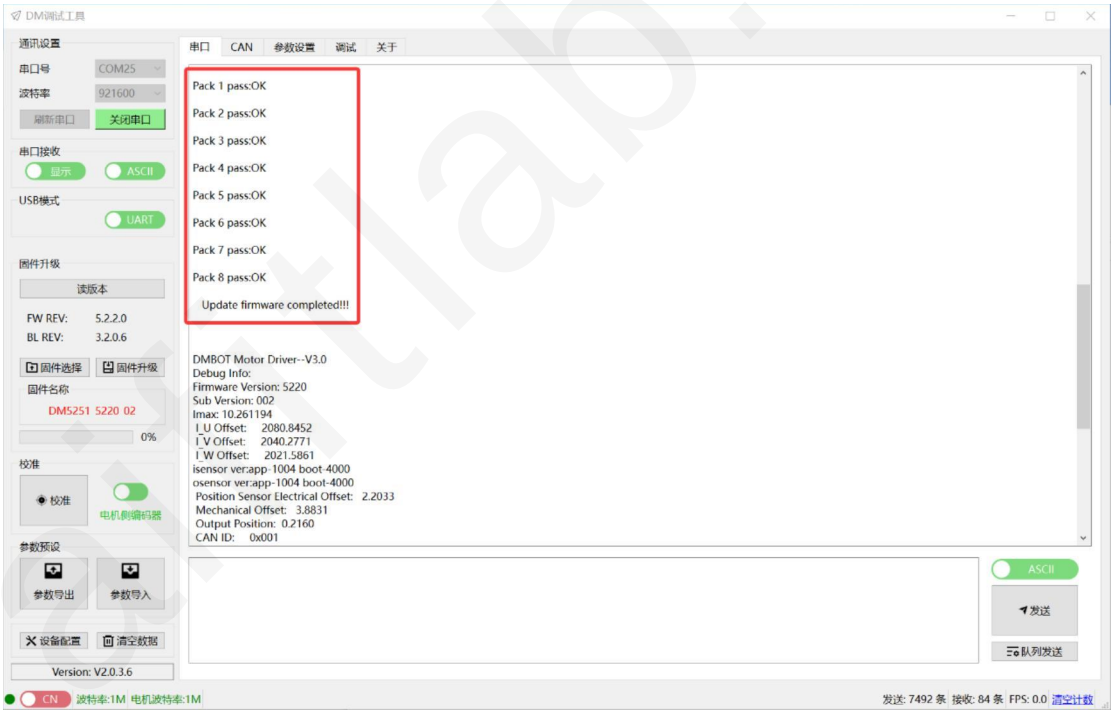
When the firmware introduces new features or receives an upgrade to fix bugs, users can perform the upgrade via the serial port to resolve the issue.

To resolve the issue, use the new feature. Connect the serial port before use, then click 'Firmware Selection' to choose the appropriate firmware.

Click to open and display the firmware name. After confirming it is correct, click the firmware upgrade button and wait for the upgrade to complete.

The upgrade is complete. You can also check the upgrade status through the serial interface.

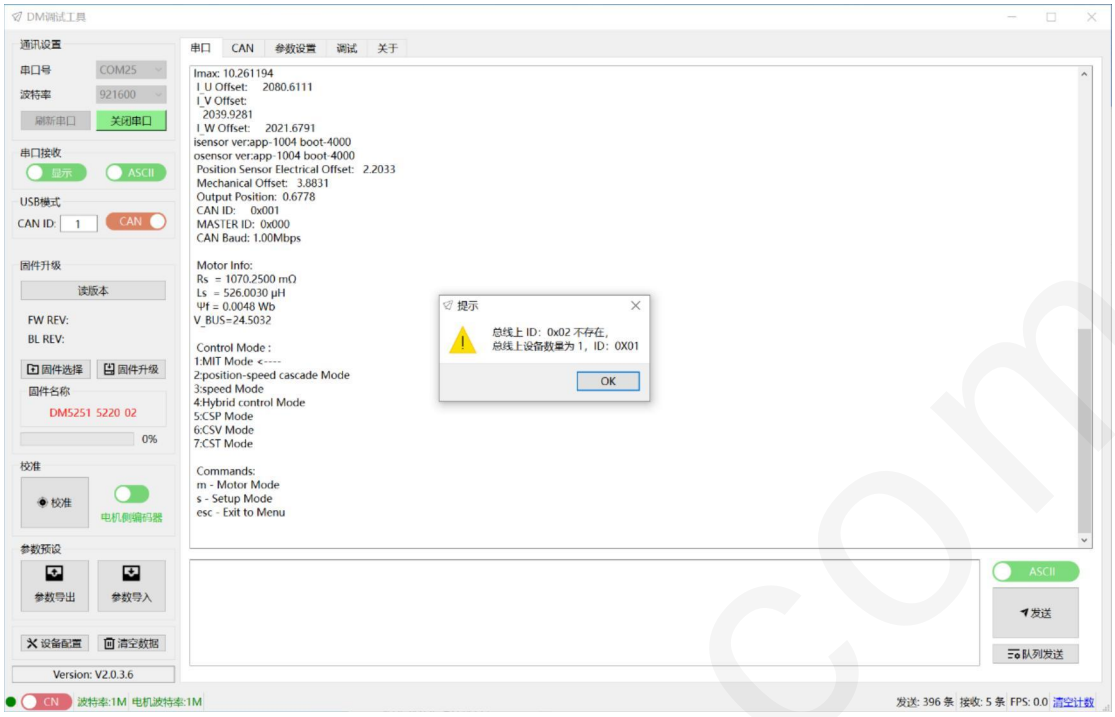




CAN firmware upgrade

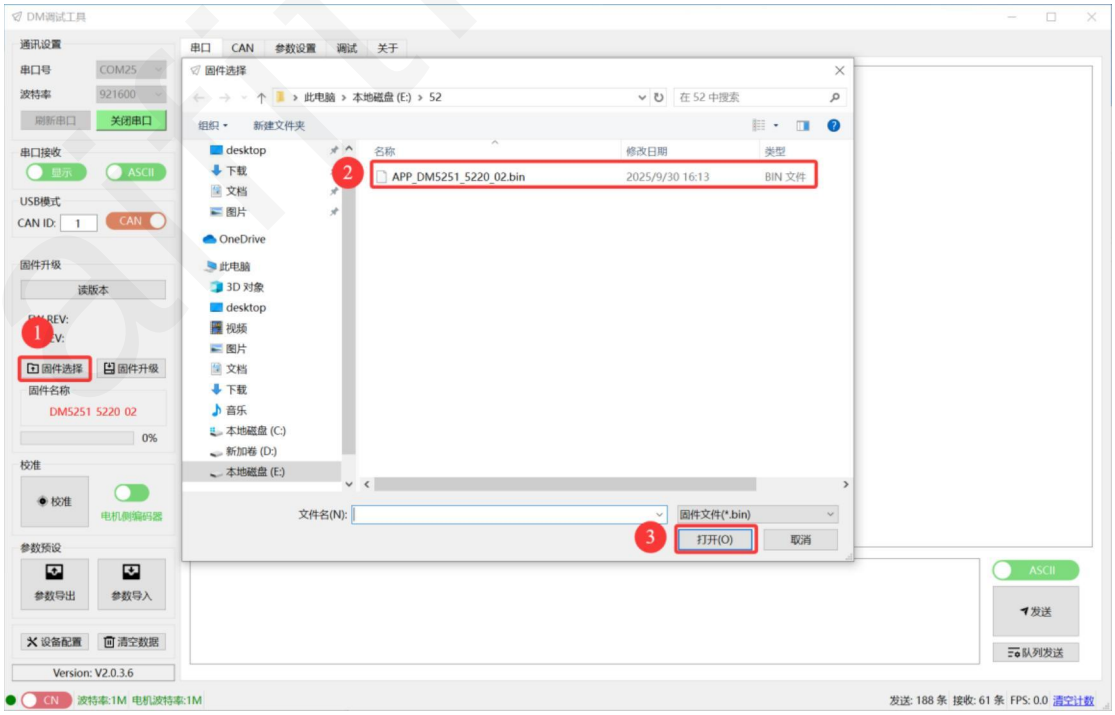
The Bootlaoder version 3.2.0.5 and later supports CAN-based firmware upgrades at the driver board's configured baud rate, with USB-to-CAN and USB-to-CAN FD conversion capabilities. Before use, connect the CAN bus to verify communication. In USB mode,

switch UART to CAN and enter the motor ID to be upgraded in the left field. If the bus is unavailable, the system will display the following error message:



No prompt is displayed if the connection is successful.

Next, click 'Firmware Selection' to choose the required firmware. After clicking 'Open', the firmware name will appear in the name field. Confirm the name is correct, then click the firmware upgrade button. Wait for the progress bar to complete, or check the upgrade status through the serial port interface.



After the upgrade is complete, the serial interface will display the corresponding prompt: