Hightorque Motor Debugging Assistant Manual

0. Applicable Version

- This manual applies to Hightorque Debugging Assistant v0.11.1 and above.

- The Hightorque Debugging Assistant supports motor firmware version v3.1.0 and later.

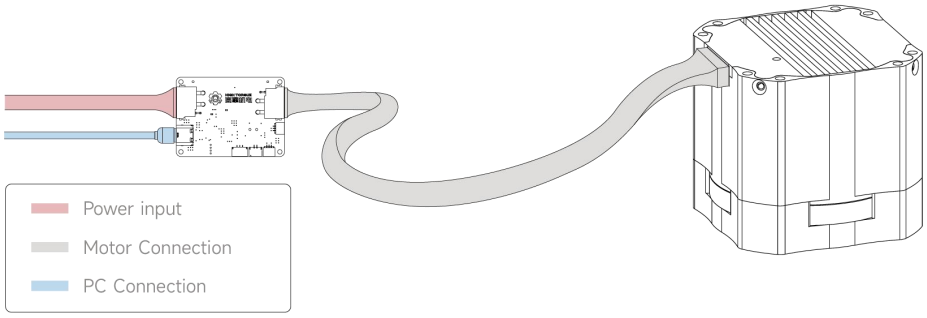
1. Motor Usage Instructions

1.1 Motor Hardware Wiring

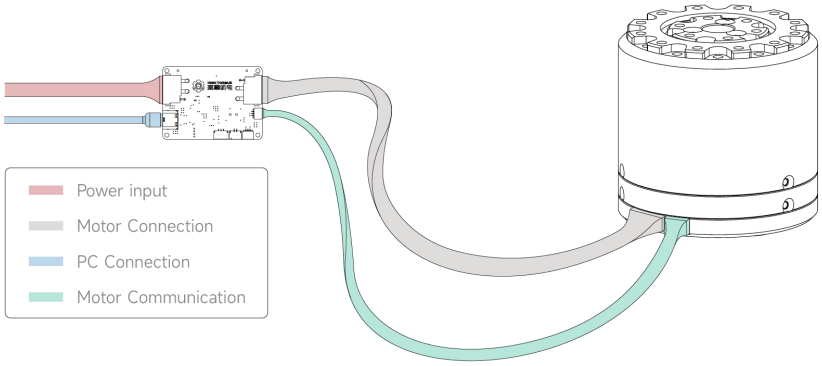
1. Rated voltage: 24VDC

2. Debugging connections:

① Connect the motor's XT30 (2+2) interface to the FDCAN module. The FDCAN module connects to the PC via its Type-C port.
- Applicable motor models: 5047, 4438



② Connect the motor's XT30 interface to the FDCAN module. The FDCAN module connects to the PC via Type-C. Use a GH1.25_3P cable to connect the motor and the FDCAN module. The FDCAN module provides power to the motor.
- Applicable motor model: 5046



2. Hightorque Motor Debugging Assistant – Usage Instructions

2.1 Basic Information and Functions

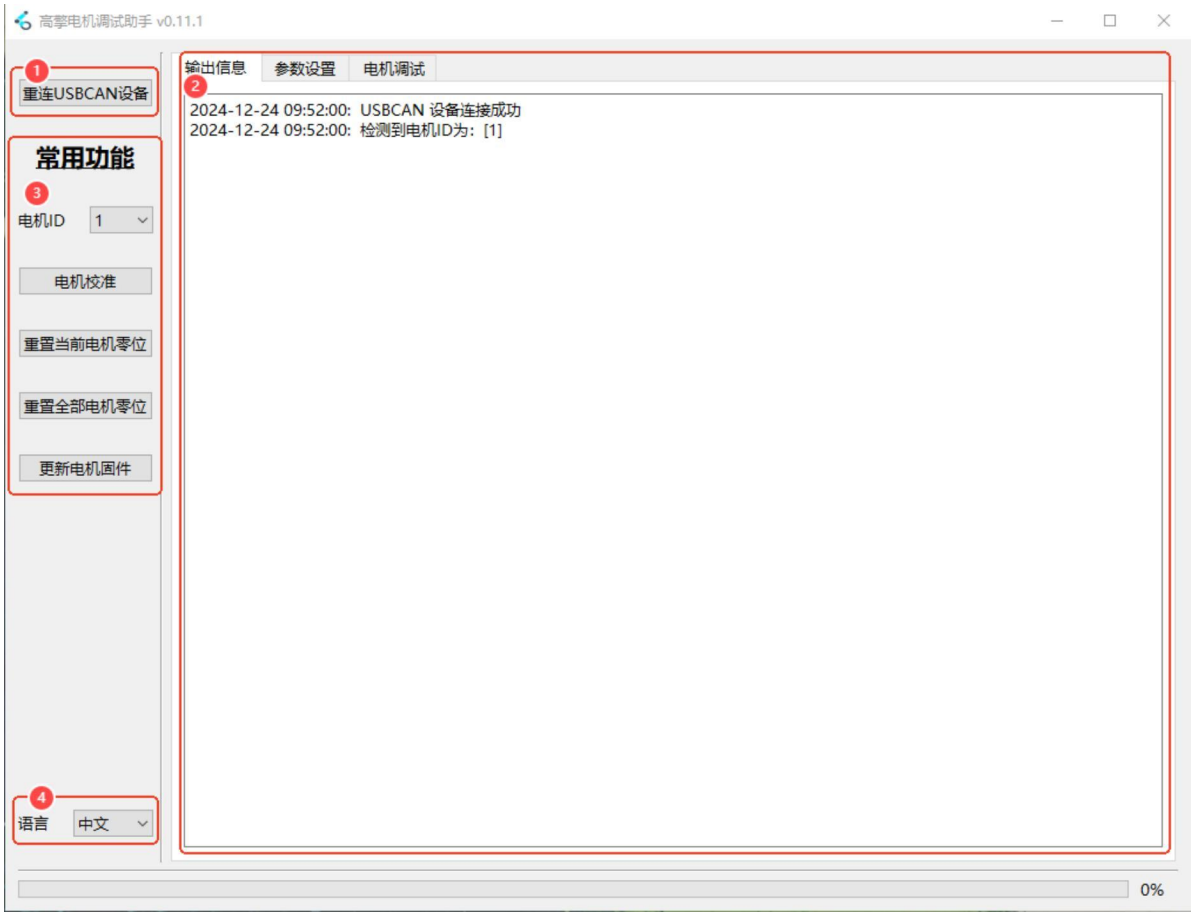
1. Reconnect USBCAN Device: Scan and connect to USBCAN devices and check for motor connectivity. If the connection is lost or multiple motors are not detected during debugging, click “Reconnect USBCAN Device” again.

2. Output Log: Displays messages including motor connection and calibration logs.

3. Common Functions:

- Motor ID: Displays the ID of the currently connected motor.
- Motor Calibration: Recalibrate the motor. Ensure the motor is not under any load during calibration. The process is logged and displayed with a progress bar.
- Reset Current Motor Zero Position: Resets the zero position for the selected “Motor ID”.
- Reset All Motors Zero Position: Resets the zero positions for all motors recognized by the host.
- Update Motor Firmware: Flash selected firmware to the motor with printed status messages.

4. Language Support: Available in Chinese and English.



2.2 Parameter Configuration

The Read Parameters and Write Parameters functions allow users to view and modify the following motor settings:

1. Basic Information:

Displays the fundamental information of the connected motor.

2. Motor Protection:

Sets protection thresholds for overvoltage and overheating. If the motor exceeds these limits, it will trigger a fault and stop operation. After modification, the new settings must be written to the motor.

3. PID Tuning:

Configures the internal PID control parameters of the motor. The i_limit parameter defines the integral limit. Any changes must be written to the motor.

4. Motor ID Configuration:

Updates the motor's ID and reinitializes motor recognition. After modification, the updated ID must be written to the motor.

5. Position Limits:

Defines the maximum and minimum position range for motor operation. If set to nan, the range is considered unlimited. Changes must be written to the motor.

6. Motion Constraints:
Sets limitations for speed, acceleration, and current during motor operation. If any value is set to nan, that parameter is considered unrestricted. Changes must be written to the motor.

2.3 Motor Debugging

1. Motor Operating Modes:

Allows testing and debugging the motor under various operation modes (see Section 2.4 for details).

2. Motor Status Monitoring:
Clicking "Add Waveform" enables real-time visualization of motor parameters including position, velocity, torque, temperature, and status code.

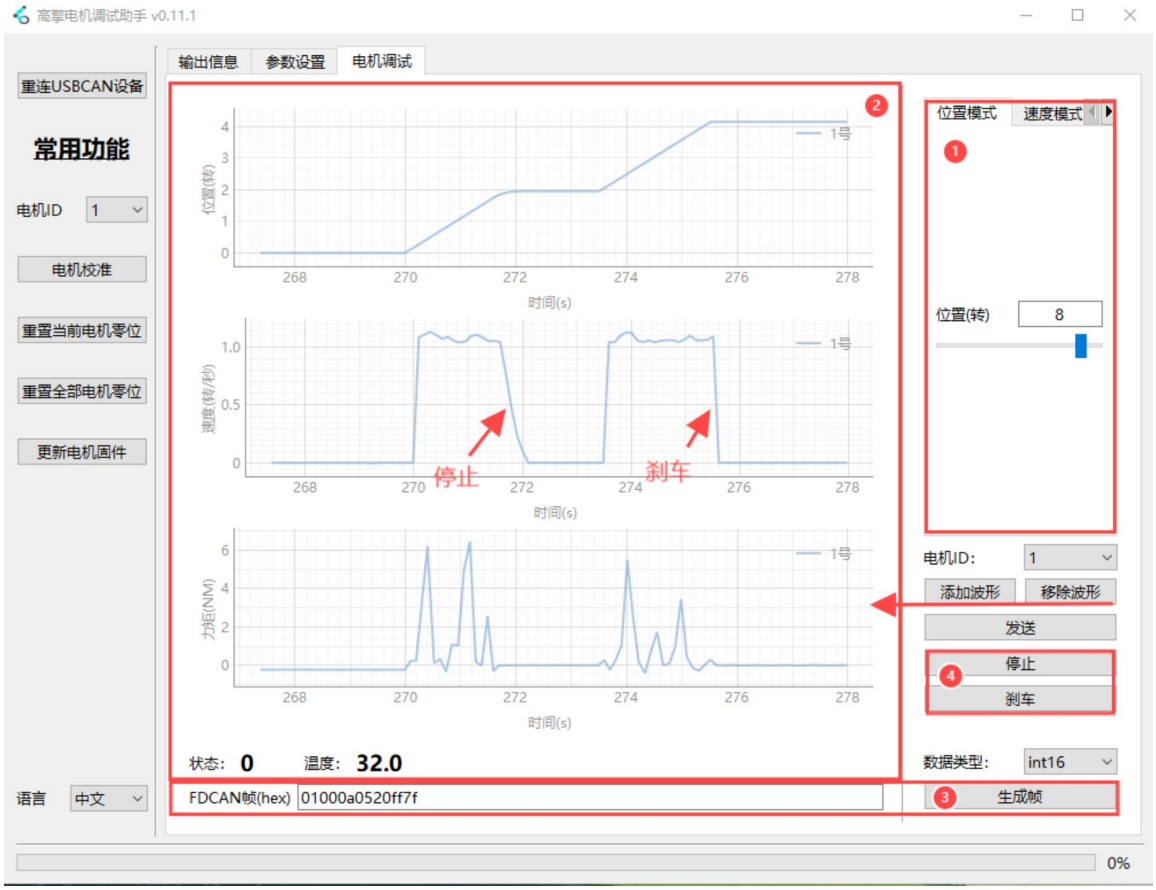
Note: A non-zero status code indicates a fault—refer to the Error Code Table in the Appendix for detailed explanation.

3. Frame Generation:

- Displays the FDCAN frame data representing the motor's current operation state. Supported formats include float, int16, int32, and raw CAN frames.
    - For float, int16, and int32 formats: Refer to the source code in CAN_H730/Src/livelybot_fdcan/livelybot_fdcan.c within the HighTorque Motor FDCAN Protocol Analysis package.(Example use cases are provided in Chapter 3: Frame Generation Examples.)
    - For raw CAN frames: Refer to:can_h730/App/livelybot/livelybot_can.c in the HighTorque Motor CAN Protocol Analysis package.Usage is similar to the FDCAN format but with protocol details specific to the CAN implementation.

4. Stop and Brake Functions:

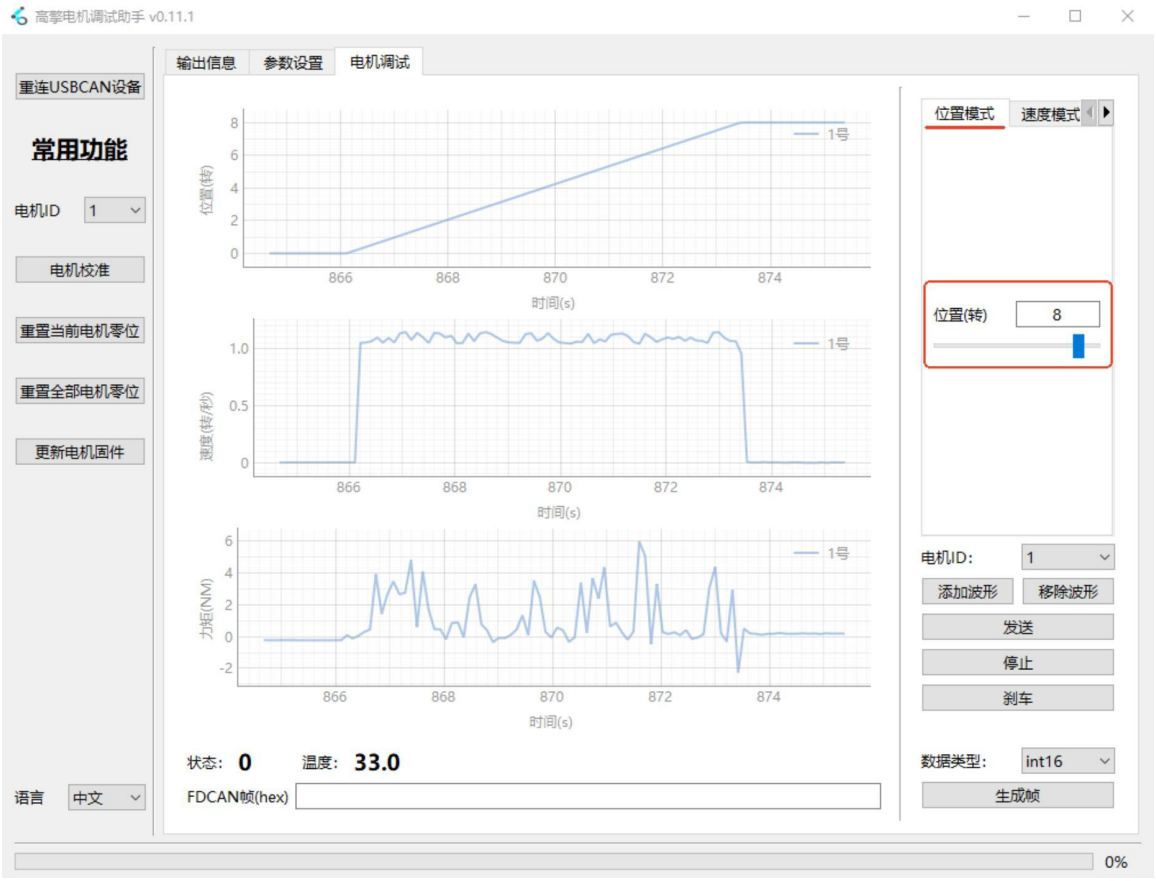
- Stop:Disconnects all three motor phases. After stopping, the motor decelerates naturally due to inertia.
- Brake:Grounds all three motor phases. The motor stops immediately when braking is applied.



2.4 Operation Modes Explained

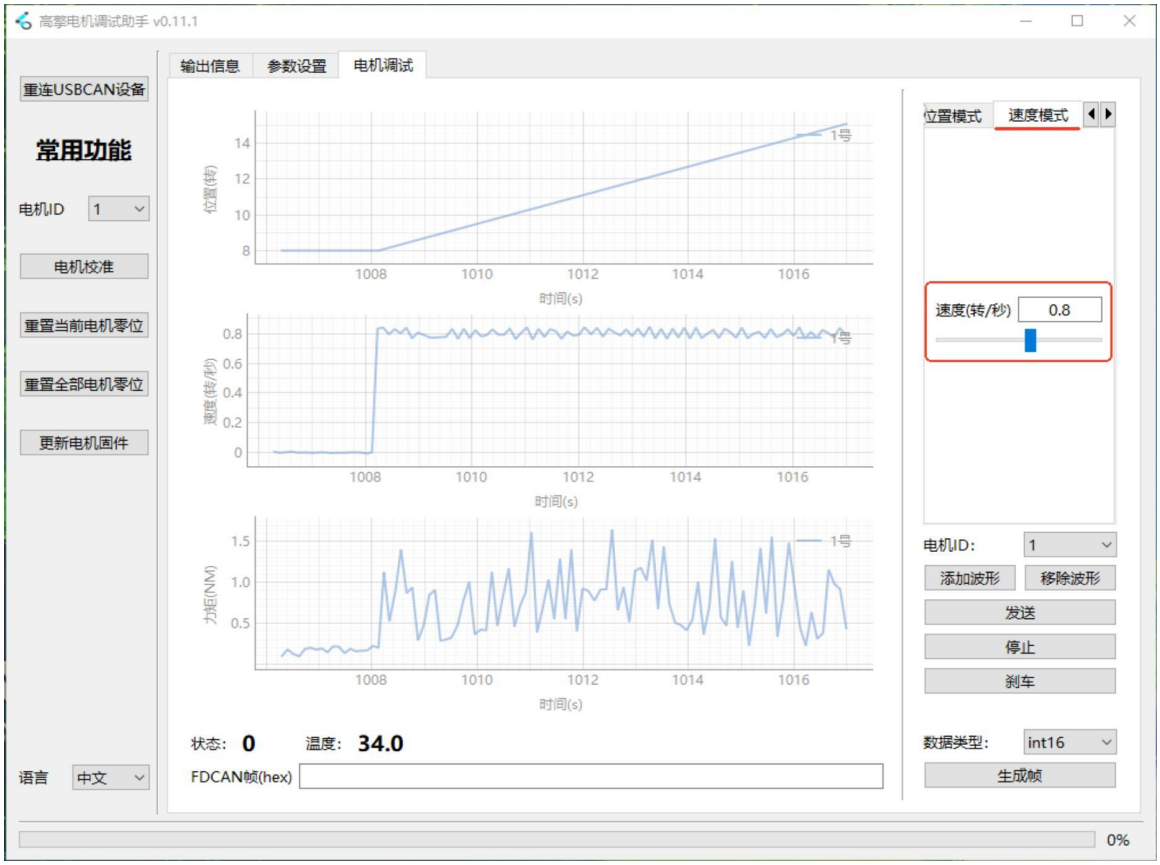
1. Position Mode:

Set the target position and click send. The motor will move to the specified position with maximum acceleration and speed. If acceleration and speed limits are set in the parameter configuration, the motor will move to the position using those specified values.



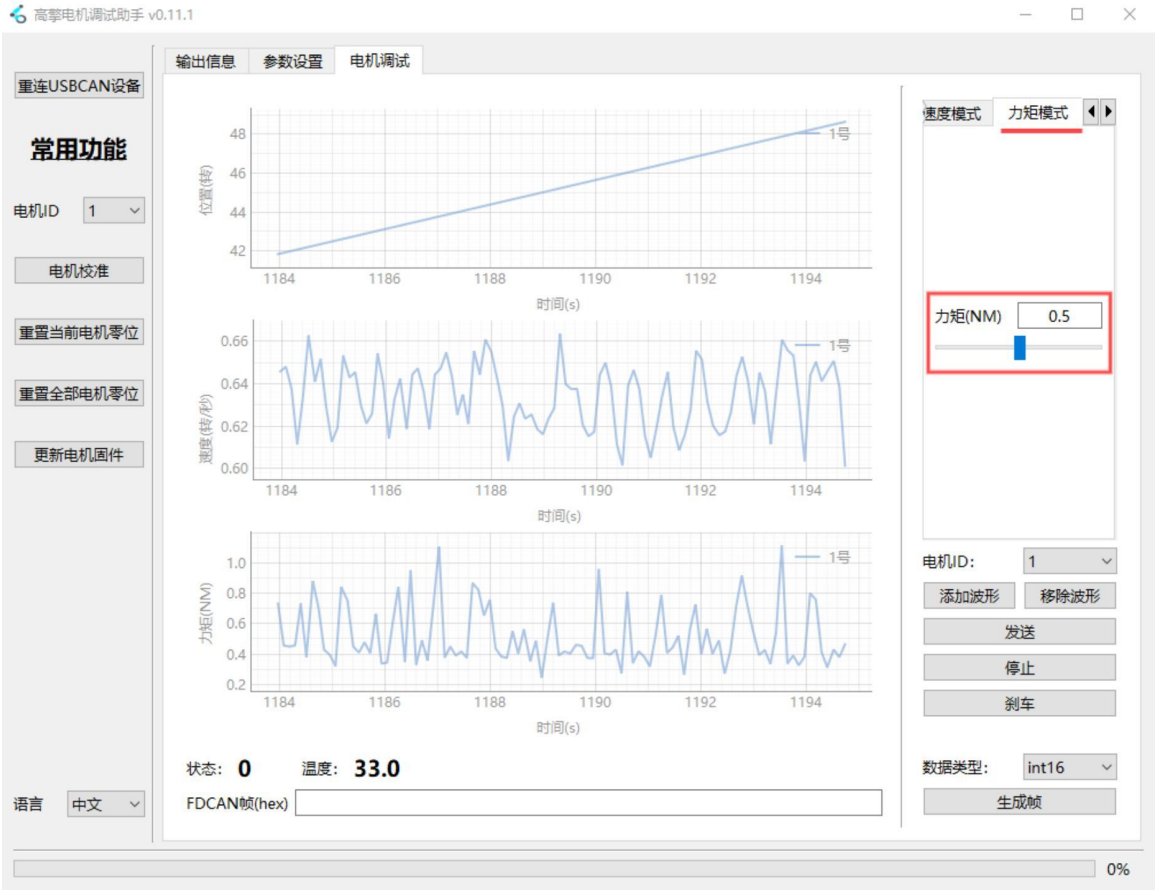
2. Speed Mode:

Set the target speed and click send. The motor will accelerate to the specified speed with maximum acceleration. If an acceleration limit is set in the parameters, the motor will use the specified acceleration to reach the target speed.



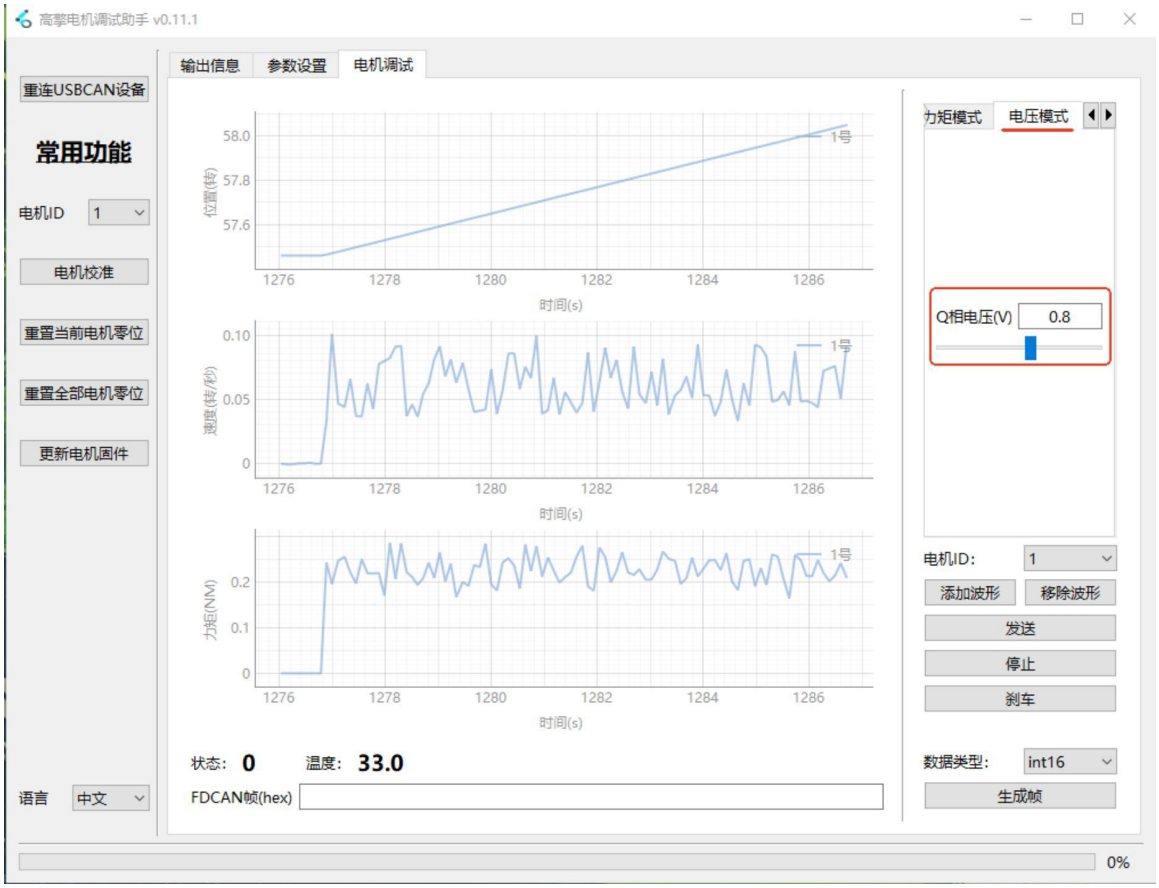
3. Torque Mode:

Set the desired torque and click send. The motor outputs the specified torque. If external resistance exceeds the set torque, the motor will stall.



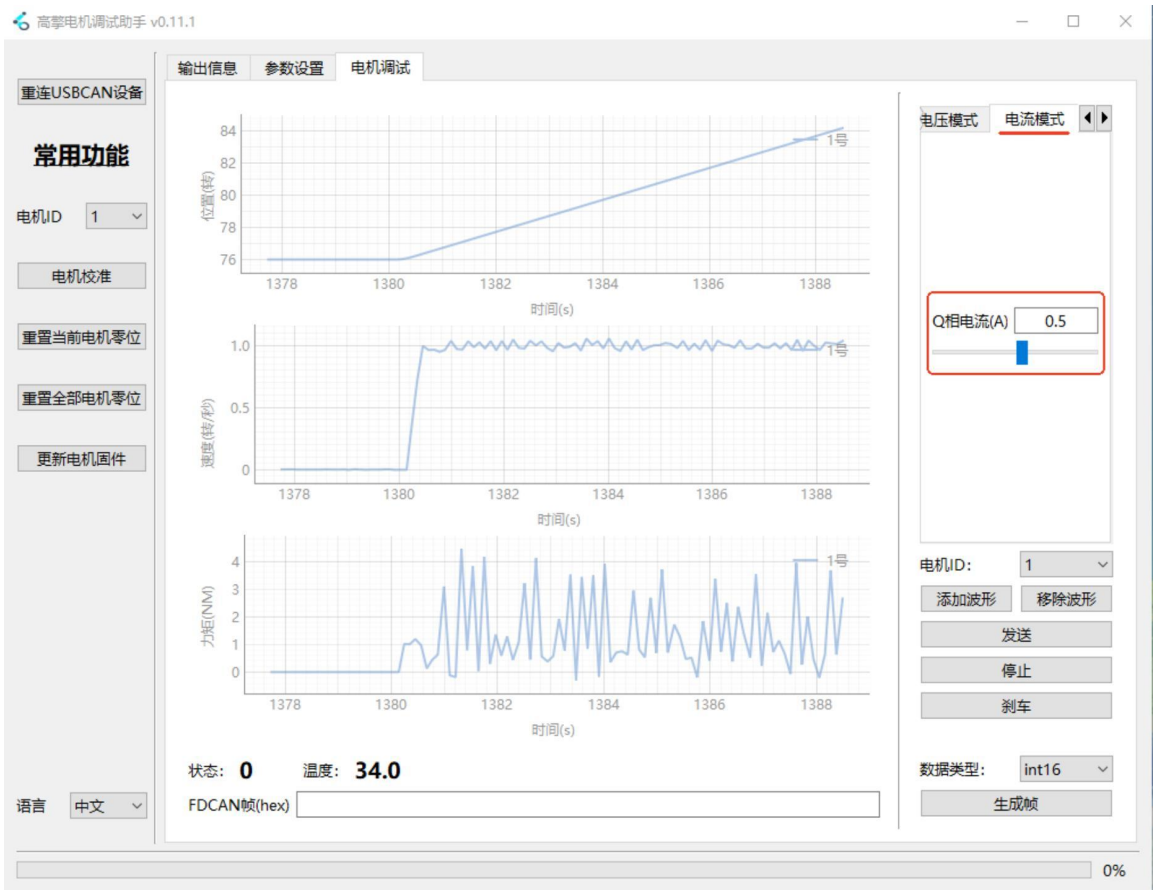
4. Voltage Mode:

Set the Q-phase voltage and send. The motor receives the specified Q-phase voltage.



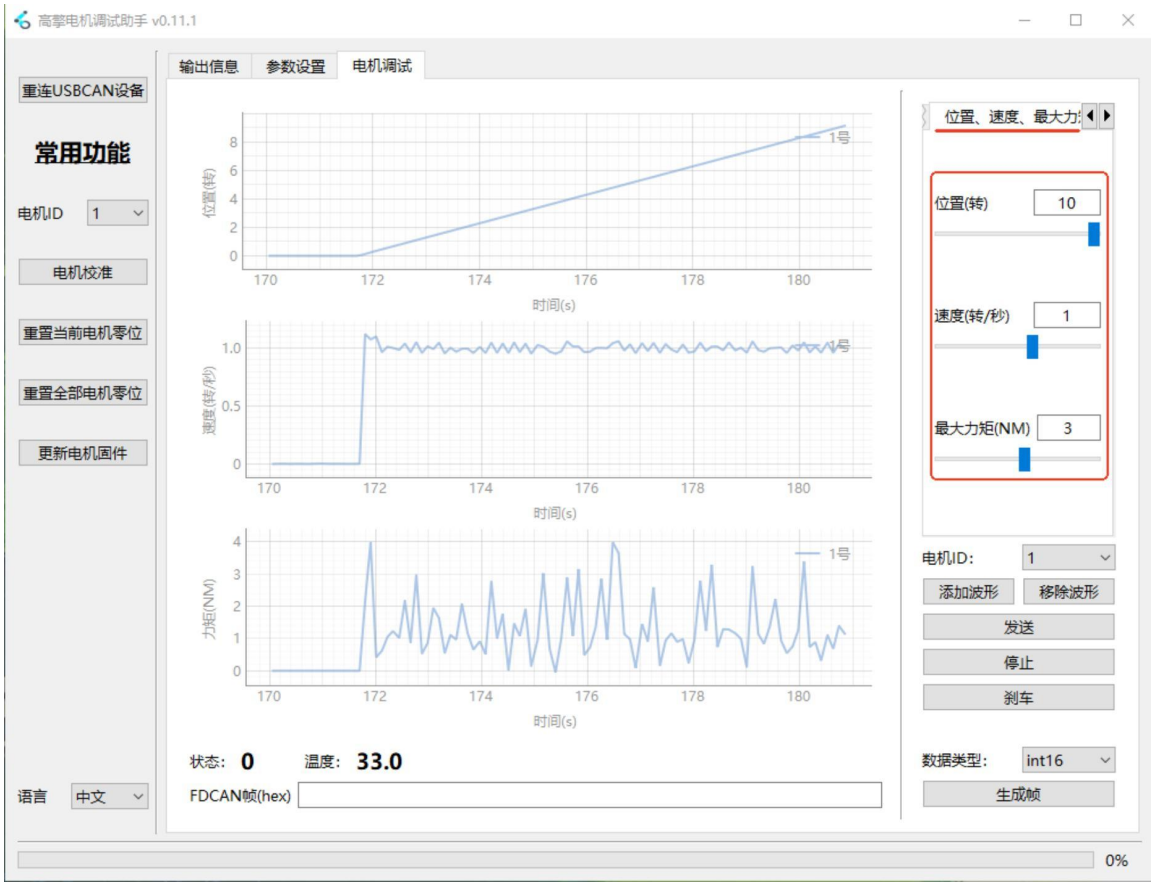
5. Current Mode:

Set the Q-phase current and send. The motor receives the specified Q-phase current.



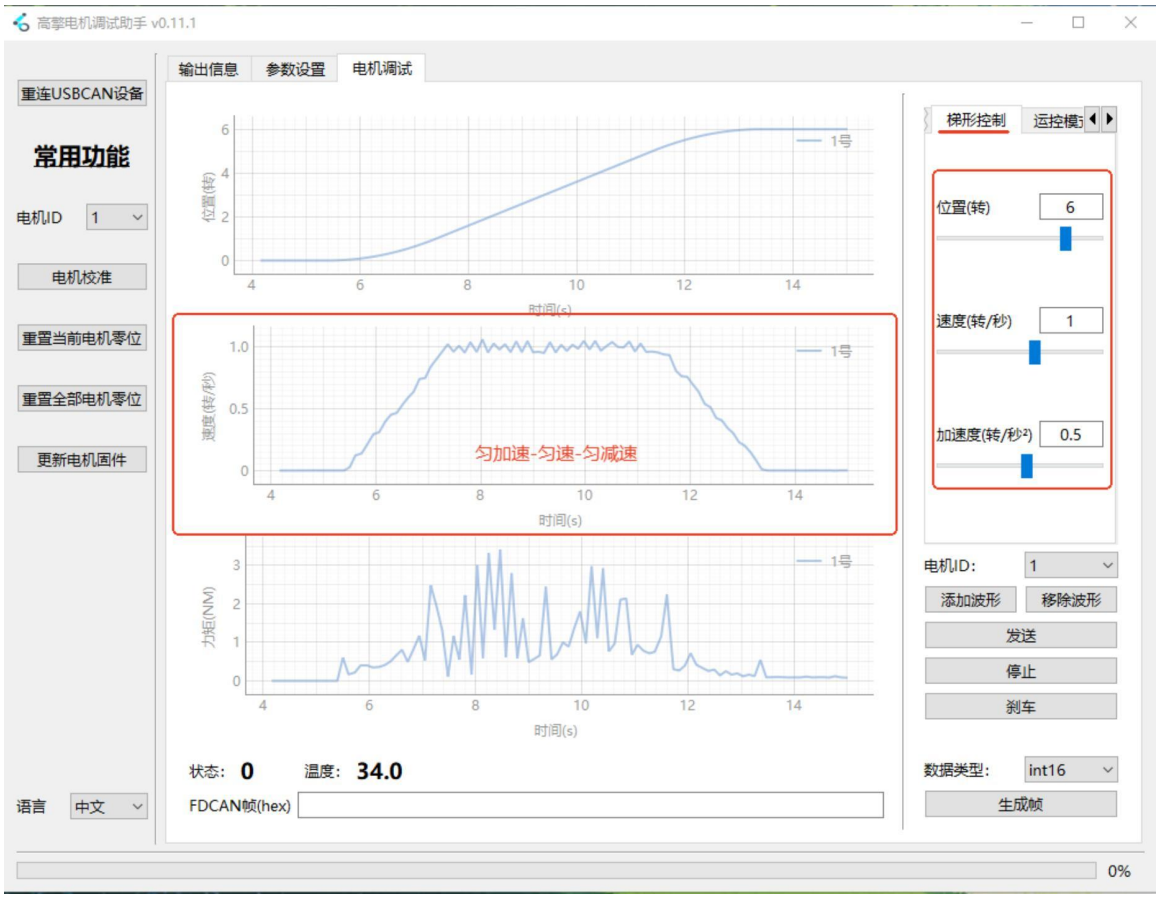
6. Position, Speed, and Maximum Torque:

The motor moves to the specified position at the specified speed and stops, while limiting the output torque. To remove torque limitation, set the maximum torque to NAN.



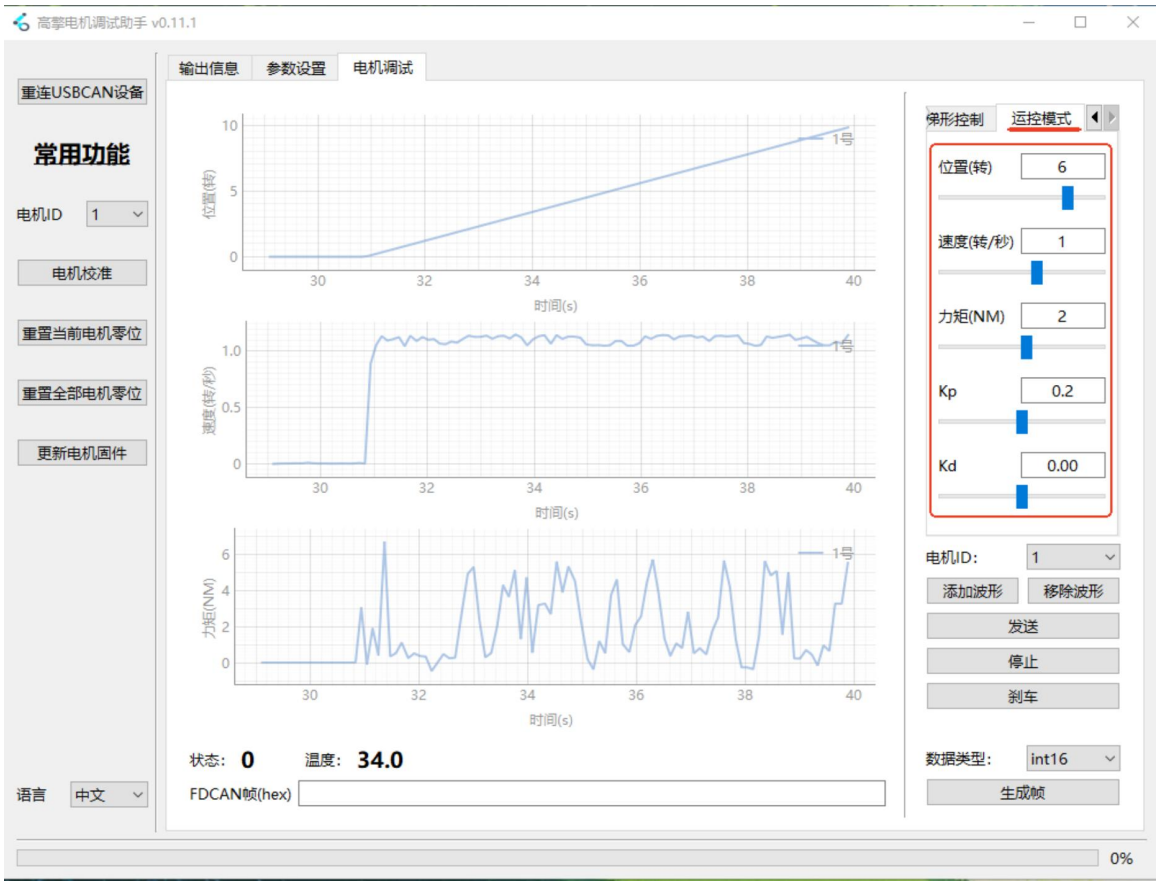
7. Trapezoidal Control:

Set the following three parameters and send. The motor will accelerate smoothly to the target speed at the specified acceleration, then run at constant speed, and finally decelerate smoothly to stop at the target position.



8. Operational Control Mode:

The motor torque output is calculated as:
Torque = Position error × Kp + Speed error × Kd + Torque offset.



3. Frame Example Analysis

For a more detailed analysis, please refer to the documents “GaoQing Motor FDCAN Protocol Analysis” including fdcan.pdf and “Register Functions, Motor Operation Modes, Error Codes, One-to-Many Mode Description.xlsx”.

Example analysis of an int32 data frame in speed mode: `01000a0a2000000080204e0000`

1. Subframe 1: Indicates that the motor enters position mode.

- `0x01`: The start byte of the first subframe

  - High nibble: `0000`, representing a write operation.

  - Low nibble:

    - High 2 bits: `00`, representing an int8_t data type.

    - Low 2 bits: `01`, indicating 1 data item.

- `0x00`: Starting register address. According to the register table, register `0x00` corresponds to motor mode setting (see the register functions and error codes spreadsheet).

- `0x0a`: Value `0x0a` is written to register `0x00`.

2. Subframe 2: Indicates unlimited position and speed set to 0.2 rotations per second.

- `0x0a`: The start byte of the second subframe

  - High nibble: `0000`, representing a write operation.

  - Low nibble:

    - High 2 bits: `10`, representing int32_t data type.

    - Low 2 bits: `10`, indicating 2 data items.

- `0x20`: Starting register address. Register `0x20` corresponds to position, and `0x21` corresponds to speed.

- `0x00 0x00 0x00 0x80`: Little-endian format, writing `0x80000000` to register `0x20`, indicating no position limit.

- `0x20 0x4e 0x00 0x00`: Little-endian format, writing `0x4e20` to register `0x21`, representing speed set to 0.2 rotations per second.

  - Converting `0x4e20` to decimal yields 20000. According to the table, an int32 value of 100000 corresponds to 1 rotation per second, so 20000 corresponds to 0.2 rotations per second (see section 2.7 of fdcan.pdf).

Appendix

1. Error Code Description Table

Please refer to the attached table for detailed error code descriptions.

Error Code	*Name*	*Description*
32	**Calibration Failure**	**During calibration, the encoder cannot detect the magnet.**
33	**Motor Drive Fault**	**Often caused by undervoltage or insufficient current.**
34	**Overvoltage**	**Bus voltage is too high.**
35	**Encoder Fault**	**Encoder reading error.**
36	**Motor Not Calibrated**	**Motor has not been calibrated yet (motors are calibrated once at factory).**
37	**PWM Period Exceeded**	**Generally due to internal firmware error.**

Error Code	*Name*	*Description*
38	Over Temperature	Exceeded maximum configured temperature.
39	Out of Limits	Attempted to start position control beyond position limits (no position limit by default).
40	Undervoltage	Voltage is too low.
41	Configuration Changed	Configuration values that require stopping were changed during operation.
42	Invalid Angle	No valid Hall encoder available.
43	Invalid Position	No valid output encoder available.