CRAF ForceControl Plugin User Guide

Original Instructions

Issue: V3.4.0

Date: 2025-05-09

SHENZHEN DOBOT CORP LTD

Disclaimer

To the maximum extent permitted by applicable law, the products described (including its hardware, software and firmware, etc.) in this document are provided **AS IS**, which may have flaws, errors or faults. Dobot makes no warranties of any kind, express or implied, including but not limited to, merchantability, satisfaction of quality, fitness for a particular purpose and non-infringement of third party rights. In no event will Dobot be liable for any special, incidental, consequential or indirect damages resulting from the use of our products and documents.

Before using our product, please thoroughly read and understand the contents of this document and related technical documents that are published online, to ensure that the robot is used on the premise of fully understanding the robot and related knowledge. Please use this document with technical guidance from professionals. Even if follow this document or any other related instructions, damages or losses may happen in the using process. Dobot shall not be considered as a guarantee regarding all security information contained in this document.

The user has the responsibility to make sure following the relevant practical laws and regulations of the country, in order that there is no significant danger in the use of the robot.

SHENZHEN DOBOT CORP LTD

Address: Room 1003, Building 2, Chongwen Garden, Nanshan iPark, Liuxian Blvd, Nanshan District, Shenzhen, Guangdong Province, China

Website: www.dobot-robots.com

Preface

Purpose

This document describes the operations of CRAF robot ForceControl plugin, making it easy for users to fully understand and use the force-control drag function.

Intended audience

This document is intended for:

- Customer Engineer
- Sales Engineer
- Installation and Commissioning Engineer
- Technical Support Engineer

Compatible versions

Product	Version
ForceControl plugin	3.4.0-stable and above
DobotStudio Pro	4.6.2.0-stable and above

Revision history

Date	Version	Revised content
2025-05-09	V3.4.0	The first release, corresponding to the ForceControl plugin 3.4.0-stable

Symbol conventions

The symbols that may be found in this document are defined as follows.

Symbol	Description
DANGER	Indicates a hazard with a high level of risk which, if not avoided, could result in death or serious injury.
WARNING	Indicates a hazard with a medium level or low level of risk which, if not avoided, could result in minor or moderate injury, robot damage.
NOTICE	Indicates a potentially hazardous situation which, if not avoided, could result in robot damage, data loss, or unanticipated result.
NOTE	Provides additional information to emphasize or supplement important points in the main text.

Contents

1. Introduction

Through the deep integration of the high-precision six-axis force sensor embedded in the CRAF tool flange with the DobotStudio Pro ForceControl plugin, the robot is equipped with sensitive "tactile feedback" capabilities. With a simple touch on the robot's end-effector, intuitive drag-teaching function is enabled, allowing the robot to automatically follow the direction of the applied force. The motion speed is proportional to the force magnitude within a certain range. Additionally, motion constraints can be set for specific directions (e.g., X/Y/Z axes). It is suitable for high-precision applications requiring "haptic interaction", such as precision assembly, medical surgical assistance, and flexible polishing.

Compared to the default joint drag, force-control drag enables more precise and smoother teaching, meeting the demands of delicate operations. The specific differences are as follows:

Drag mode	Force-control drag	Joint drag
Implementation method	Based on the force sensor	Based on the current loop of joints
Force-applied part	Only by applying the force on the end tool can the robot be dragged, and the joint cannot be dragged	Any part of the robot is effective when force is applied, and a joint can be dragged separately
Positioning accuracy	High	Lower

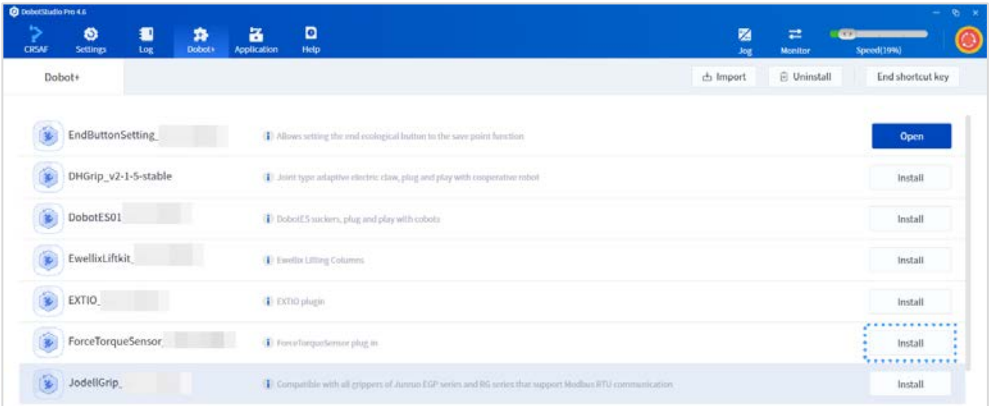
Moreover, by presetting collision thresholds (force/torque), the force sensor enables collision detection, triggering emergency stops or compliant evasive actions within milliseconds in the event of unexpected external forces, ensuring safe human-robot collaboration. If you need to realize the control of force-control drag or force sensor collision detection through secondary development, refer to the *Dobot TCP_IP Remote Control Interface Guide V4.6.0*.

2. Installation and Connection

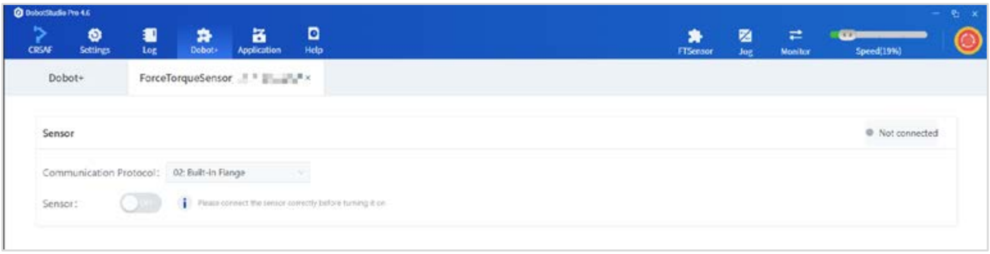
1. After connecting to the robot using DobotStudio Pro, navigate to the **Dobot+** page.

> **NOTE**
>
> The CRAF ForceControl plugin requires DobotStudio Pro 4.6.2.0-stable and above.

2. Click the "Install" button next to **ForceTorqueSensor** to install the ForceControl plugin.

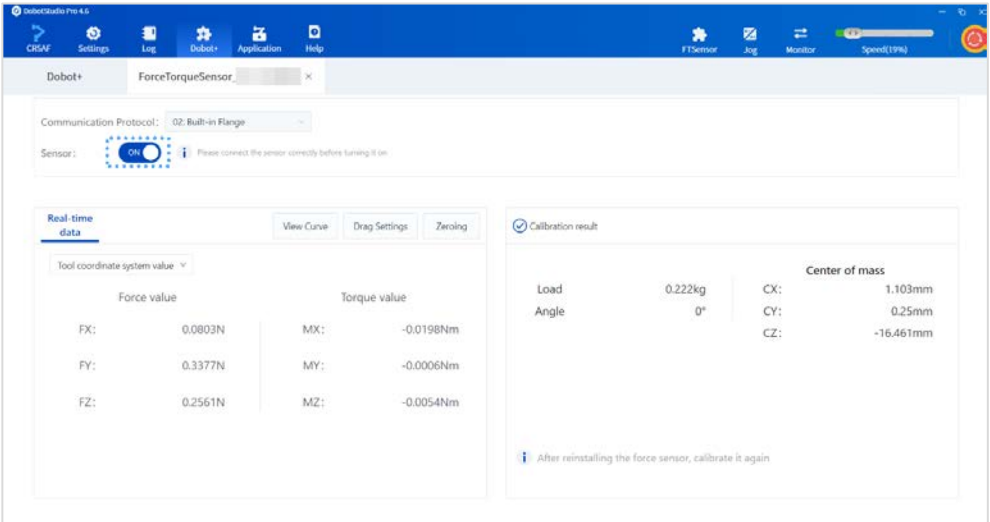


3. Once installation is complete, the system will automatically navigate to the page of the ForceControl plugin.



> **NOTE**
>
> To uninstall the plugin, select the ForceControl plugin and click "**Uninstall**" on the **Dobot+** page. To install a different version of the ForceControl plugin, first uninstall the current one, then import and install the desired version.

4. Turn on the sensor switch, and the connection status of the sensor will be displayed at the upper right corner of the interface.



Status	Color	Description
Disconnected	Gray	Sensor data cannot be obtained
Connected	Green	Sensor data can be obtained normally
Data abnormalities	Red	Data can be obtained, but the data is abnormal
Low frequency	Red	Data can be obtained, but the frequency of data reporting is too low

NOTE

When DobotStudio Pro is connected to a CRAF series robot, the sensor's **communication protocol** defaults to "02: Flange-Embedded" and cannot be modified.

3. ForceControl Plugin Installation

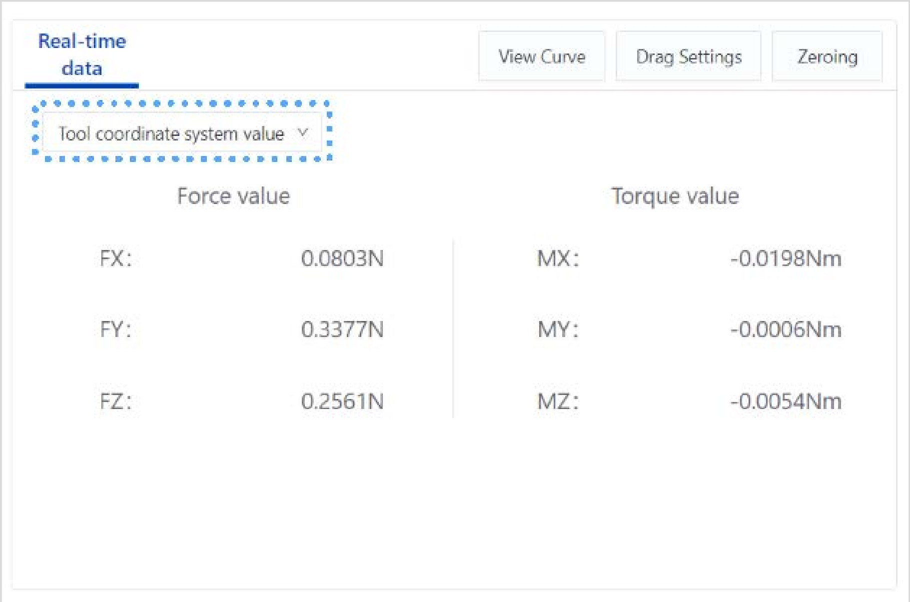
3.1 Real-time data

Real-time data means (the value read by the force sensor) - (force variations caused by load parameters). It is recommended to configure the load parameters using the automatic load identification function. For detail operations, refer to *DobotStudio Pro User Guide*.

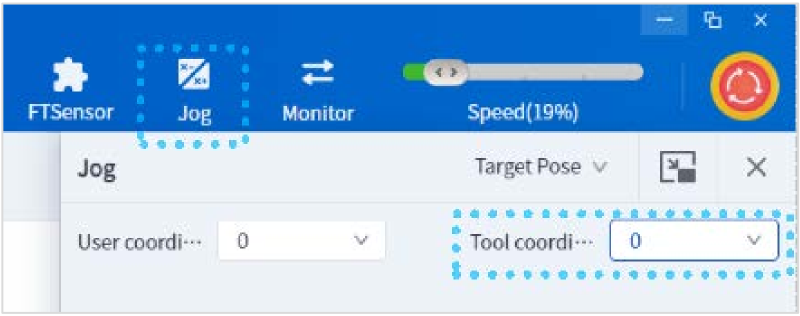
NOTICE

After configuring the load parameters, the robot's end posture can be freely adjusted. Monitor real-time data through the ForceControl plugin to check for significant changes and determine whether the load identification results are accurate.

You can modify the reference coordinate system for the real-time data via the **drop-down box**.

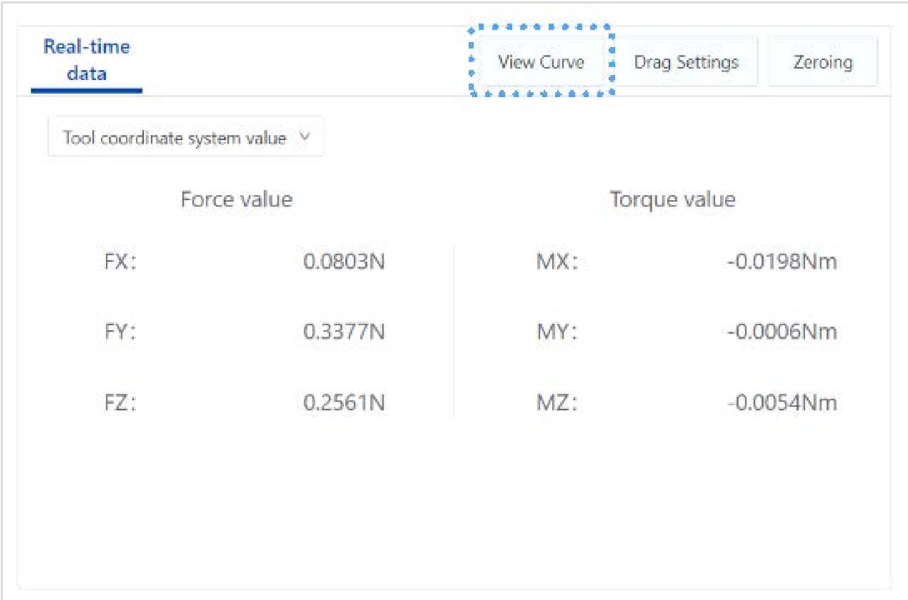


- **Sensor coordinate system:** Displays force and torque values based on the six-axis force sensor coordinate system. The six-axis force coordinate system coincides with the TCP, with its origin located at the flange center point of the TCP. The force/ torque directions follow the right-hand rule.
- **Tool coordinate system:** Displays force and torque values based on the current tool coordinate system. To change the tool coordinate system, set it in the **Jog** panel.

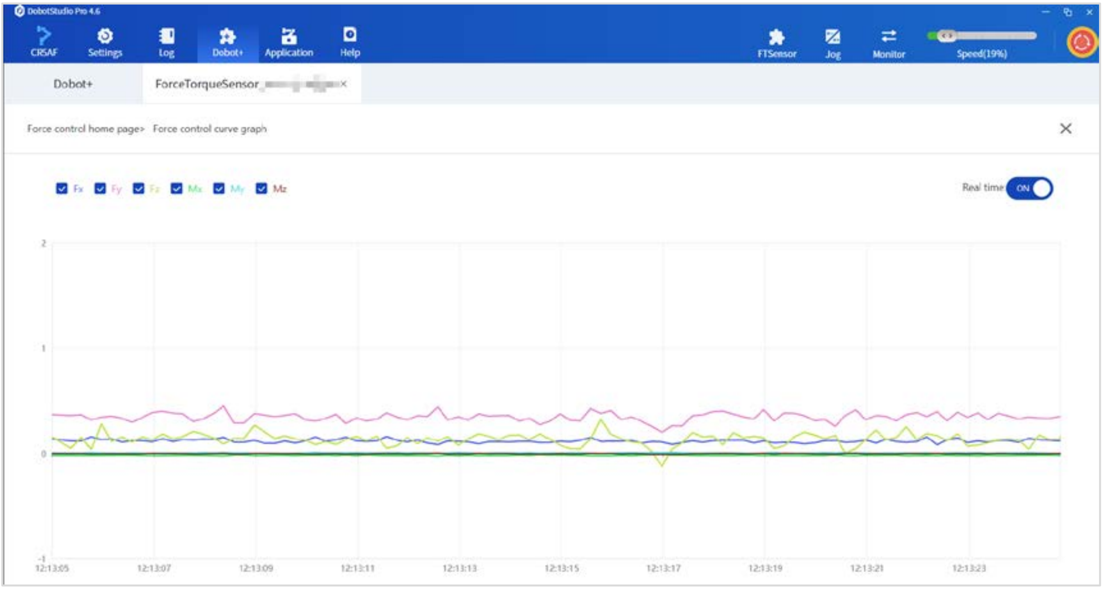


3.2 View curve

Click the **"View curve"** button to display real-time force control curves for each direction.



The force control curve displays up to 100 sets of data, with an update frequency of 200ms. Clicking any data point on the curve allows you to view the corresponding force and torque values.



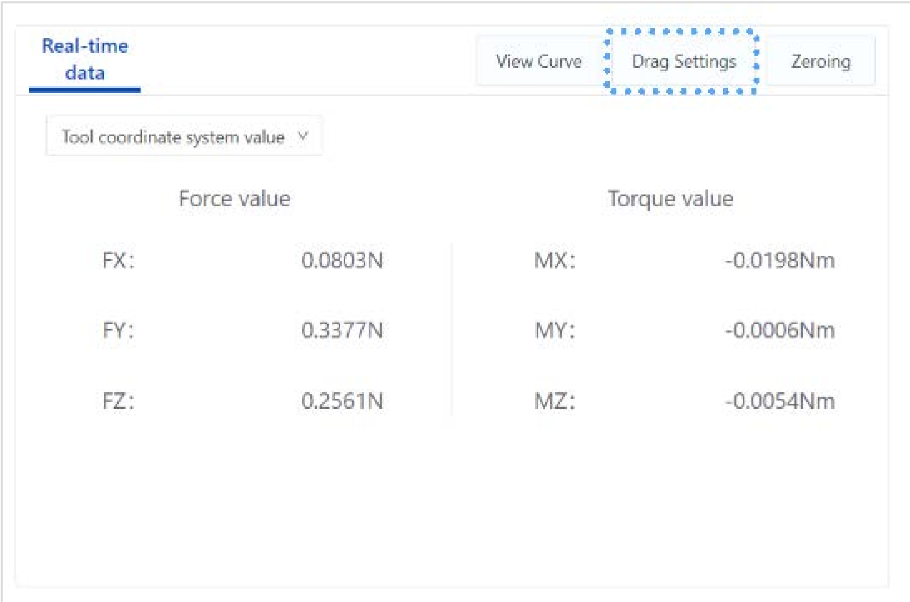
Real-time switch: Turning on the real-time switch automatically obtains the current controller time and updates the curve in real-time. When turned off, the updates pause, allowing you to check the curve.

Direction switch: Tick the direction (Fx, Fy, Fz, Mx, My, Mz) to display the corresponding force control curve for that direction. By default, all curves are displayed.

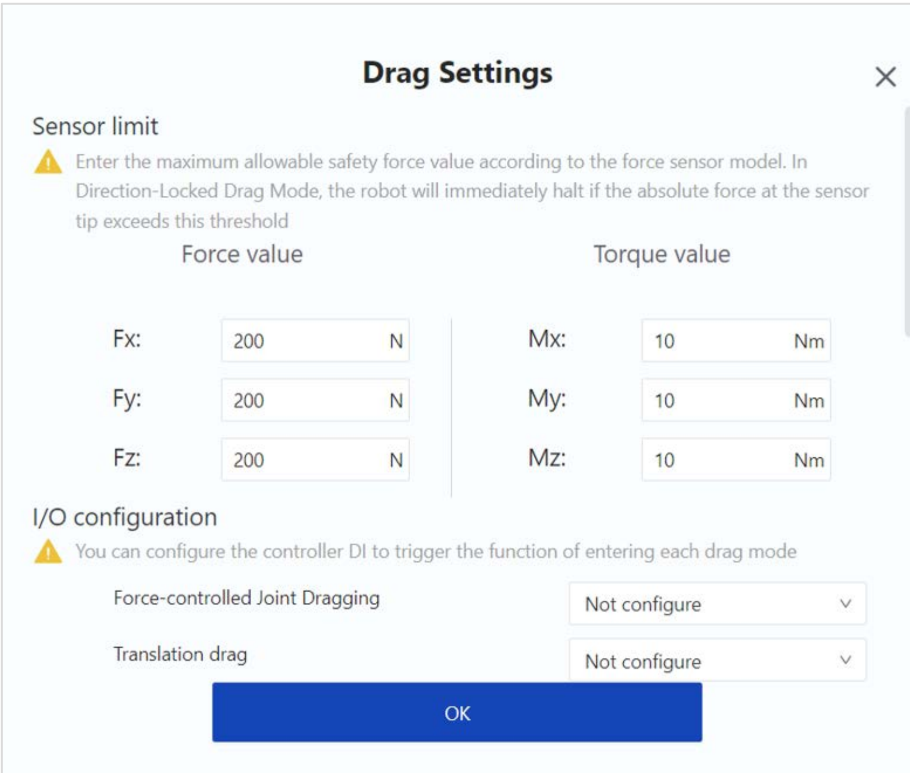
After reconnecting the robot to DobotStudio Pro, return it to default switch settings (Disabled).

3.3 Drag settings

You can click the **"Drag settings"** button to enter the sensor dragging configuration page.

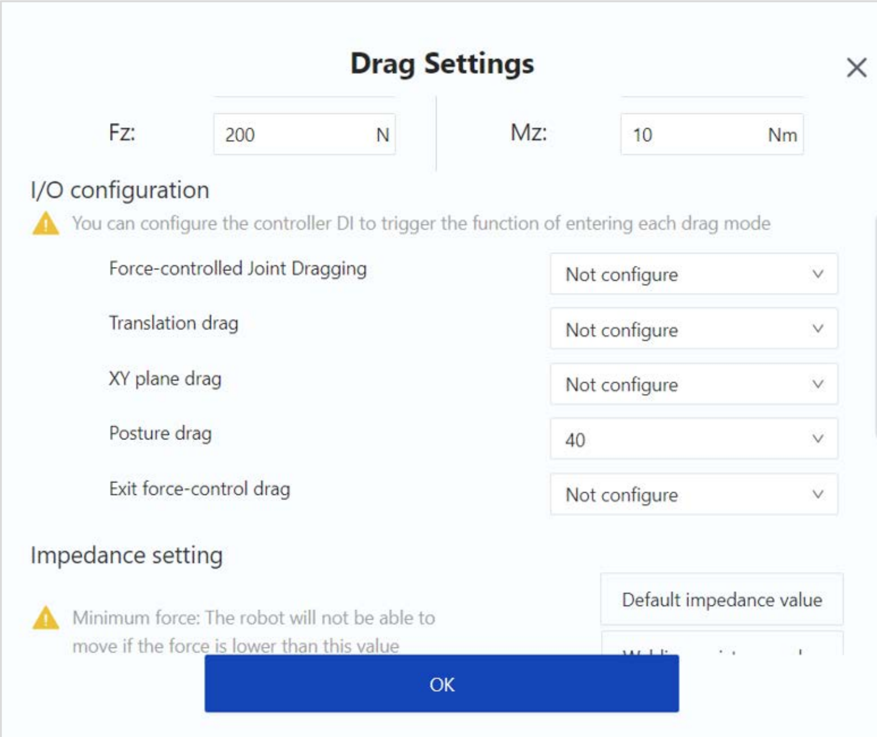


3.3.1 Sensor limit



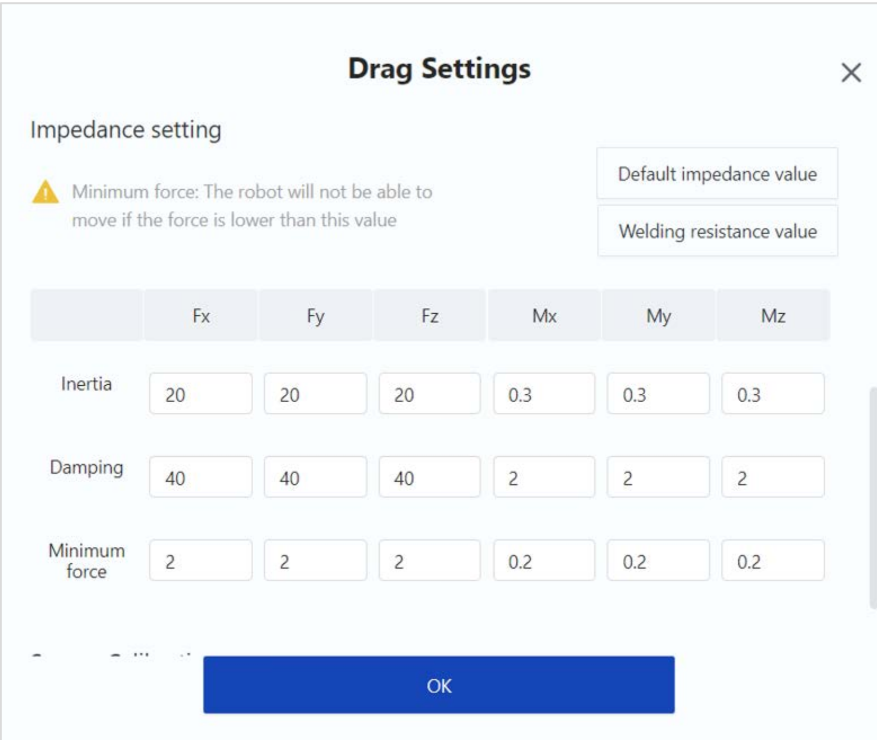
You can set the maximum safe force value and torque value for the sensor limit as needed. During force-control dragging, if the absolute value of the detected force or torque exceeds the set limit, the robot will trigger an alarm and stop its motion.

3.3.2 I/O configuration



After configuration, you can trigger the controller DI to enter or exit each drag mode of the force control function. For mode descriptions, please refer to Drag mode.

3.3.3 Impedance settings



Set the impedance values for force-control drag.

- Inertia: Range (0, 1000].
- Damping: Range [0, 1000].
- Minimum force: Range [0, 200]. If the dragging resistance feels excessive, you can reduce the minimum force threshold; However, it is recommended not to set the minimum force below 2N, as this may pose risks during operation.

Click **"Default impedance value"** to automatically fill in the recommended impedance values for general scenarios.

Click **"Welding impedance value"** to automatically fill in the recommended impedance values for welding scenarios.

NOTICE

- It is not recommended to modify the inertia and damping parameters. Simply select "Default impedance value" or "Welding impedance value" according to your application scenario to use the preset parameters.
- These parameters in impedance settings cannot be modified while the robot is in motion.

3.3.4 Sensor calibration

The built-in six-axis force sensor in the CRAF series tool flange has undergone precise calibration at the factory and generally does not require recalibration under normal operating conditions. If abnormal data is detected during actual use, calibration can be performed under the guidance of technical support.

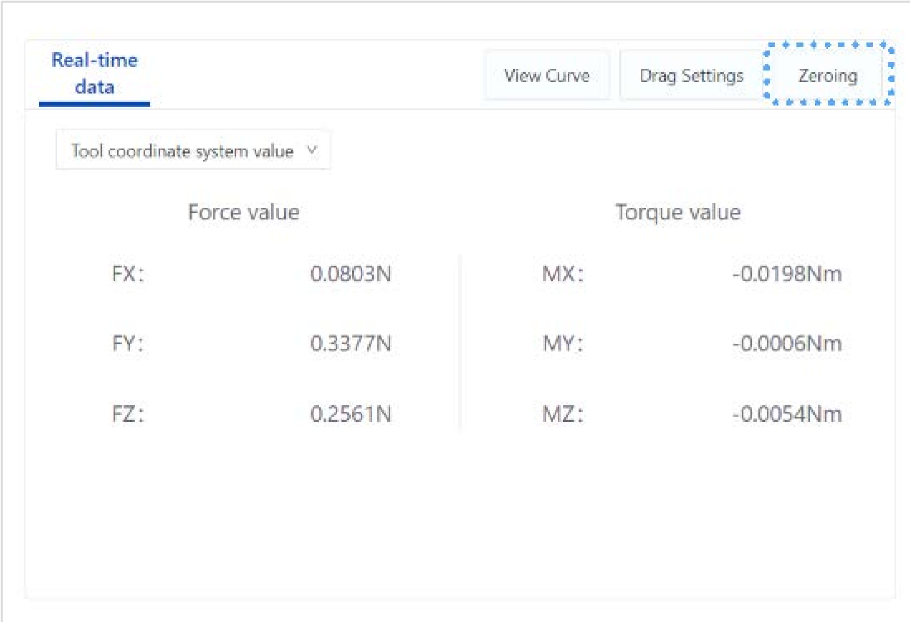
NOTE

Calibration cannot be performed if the sensor is offline, in an abnormal status, or in motion.

3.4 Zeroing

Zeroing refers to the operation where the robot compensates for the initial offset value of the force sensor, allowing the system to identify the "mechanical reference zero point" under the current load parameter configuration.

Force sensors inevitably experience temperature drift and zero drift. Before using force control functions, it is recommended to check whether the displayed force and torque values are accurate. If deviations are detected, the zeroing command must be executed to ensure the accuracy of subsequent sensor data.



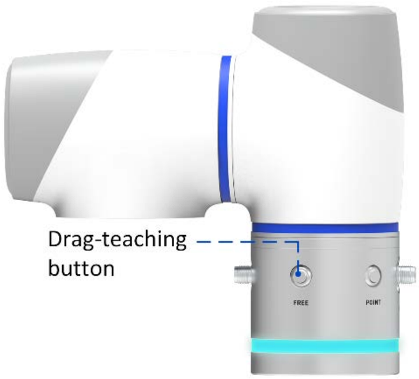
Click **"Zeroing"** to set the current force and torque values of the sensor to 0. After zeroing, you can change the robot's posture and observe whether the real-time data shows significant fluctuations. Under normal circumstances, without dragging the robot, the force value should not exceed 10N and the torque value should not exceed 2N·m. Otherwise, recalibration is required.

4. Force-Control Drag

4.1 Enter force-control drag

1. Click  on the top toolbar of DobotStudio Pro to enter the force-control drag panel, and click it again to exit the panel.
2. When the robot is enabled, press and hold the drag-teaching button to activate force-control drag mode; release the button to exit this mode.

Drag-teaching button (except CR20AF)

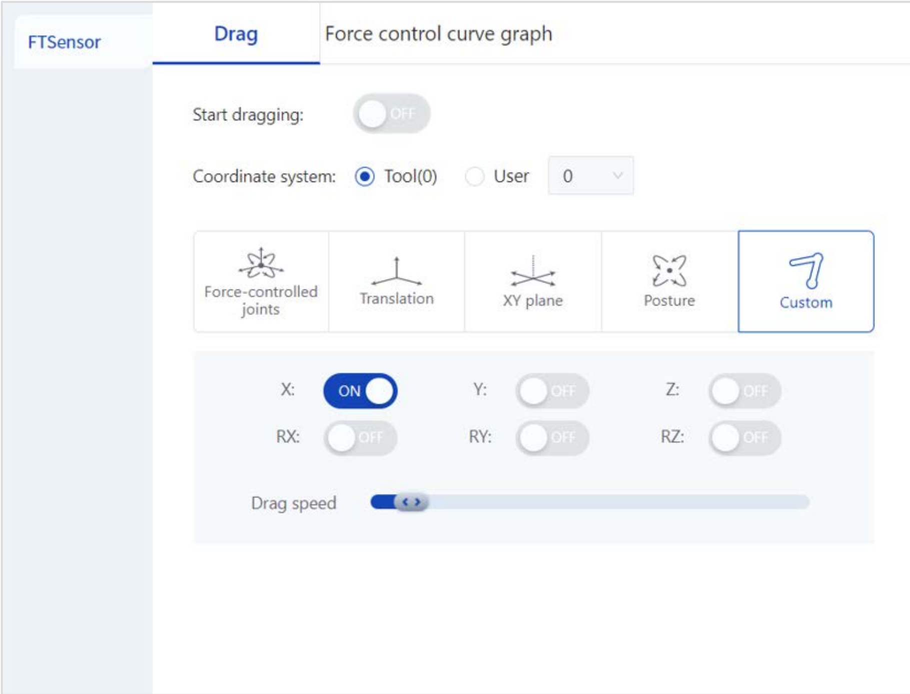


Drag-teaching button (CR20AF)

NOTE

- When entering force-control drag, the previously set dragging options will be retained.
- The drag switch  on the top toolbar and the drag-teaching button on the tool flange affect each other. If you enter the drag mode by clicking the switch on the top toolbar, then pressing and releasing the drag-teaching button on the tool flange will cause the robot to automatically exit the drag mode.

4.2 Force-control drag

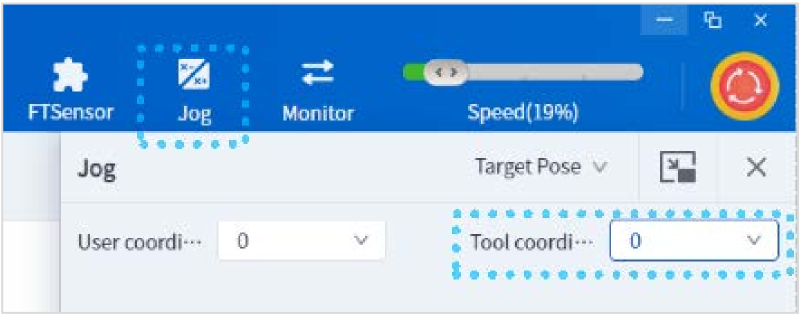


4.2.1 Drag

- **Start dragging:** When the robot is enabled and in a stationary status, you can turn on this switch to enter the force-control drag mode.

  NOTICE
  - During force-control dragging, monitor the real-time data from the force sensor. If the real-time data is abnormal, issues such as unintended robot movement or uncontrolled dragging may occur.
  - When dragging, it is recommended that the holding point be about 200mm from the center of the six-axis force sensor. The farther the distance, the more obvious the leverage effect on the posture, and jitter may occur.
  - During dragging, if you approach the joint limits or the boundaries of the safety wall/safety area, the dragging resistance will significantly increase. Then, you should change the dragging direction to avoid triggering an alarm.

- **Coordinate system:** Set the reference coordinate system for force-control dragging.
  - If the reference coordinate system is a user coordinate system, select the user coordinate system index here.
  - If the reference coordinate system is a tool coordinate system, it defaults to the current tool coordinate system. To change the tool coordinate system, set it in the Jog panel.



NOTE

Regardless of whether force-control dragging references the user coordinate system or the tool coordinate system, dragging in the Rx, Ry, and Rz directions is always based on rotations in the tool coordinate system.

1. **Drag mode:** Set the allowable directions for movement during force-control dragging.

Drag mode	Description
ForceControl Joint	Drag freely in any axis direction. The center point of posture rotation is the flange center point.
Translation	Only translate within space along the axes of the reference coordinate system. Equivalent to enabling X, Y and Z directions in custom mode.
XY plane	Only translate in the direction of the XY plane of the reference coordinate system. Equivalent to enabling X and Y directions in custom mode.
Posture	Only rotate within space. Equivalent to enabling RX, RY and RZ directions in custom mode. The center point of rotation is the origin of the current tool coordinate system.
Custom	Restrict movement in one or several directions. When all directions are enabled, it differs from ForceControl Joint mode as the center point of posture rotation is the origin of the current tool coordinate system.

2. **Drag speed:** Set the speed for force-control dragging. The higher the speed, the lighter the drag sensation.

NOTE

The maximum speed for force-control dragging in X/Y/Z directions is 0.3 m/s, and in RX/RY/RZ directions is 85.94°/s. When the maximum speed is reached, the drag sensation becomes heavier to prevent over-speeding.

4.2.2 Force control curve

By clicking the **"Force control curve"** button on the force-control drag panel, you can view the real-time force control curves for each direction. For details, see View curve.

5. Force Sensor Collision Detection

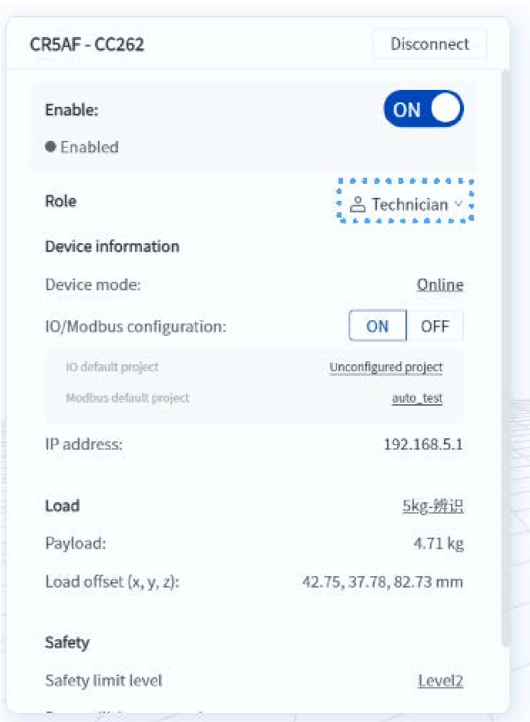
The CRAF series robots can detect external collisions via the built-in six-axis force sensor at the end effector. When the collision force or torque on the robot's end exceeds user-defined thresholds for collision force or torque, the robot will either pause or stop its movement based on the preset configuration.

In special cases, both joint collision and force sensor collision may be triggered simultaneously. The differences between them are as follows:

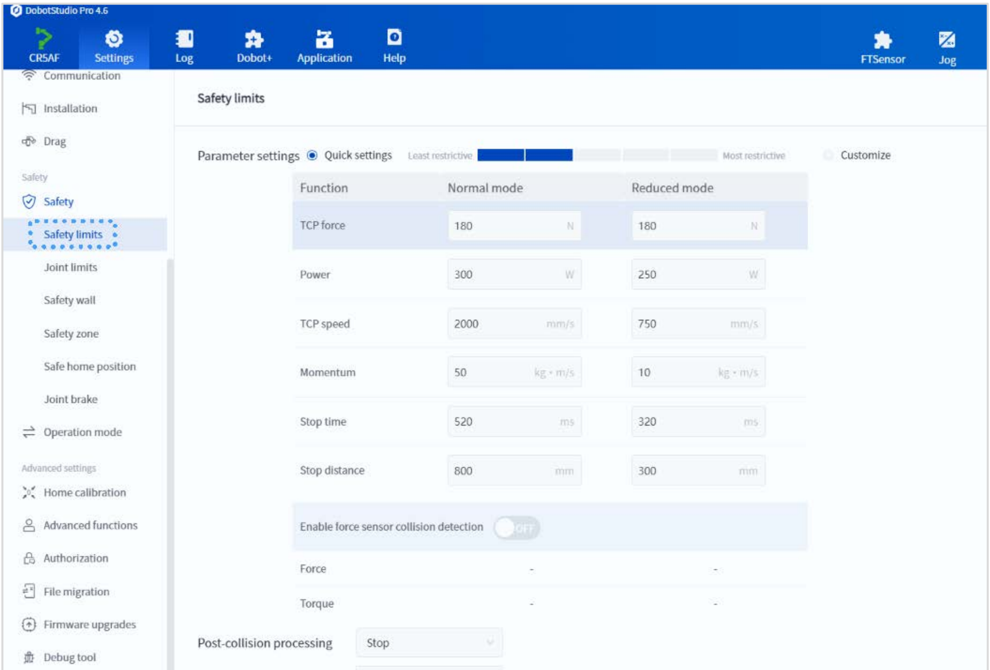
Collision detection mode	Joint collision	Force sensor collision
Switch position	Joint collision detection cannot be disabled.	Safety limits page
Trigger collision position	Based on joint current calculation and detection, collisions can be triggered at any position from J1 to J6	Based on end-effector force sensor data calculation and detection, collisions can only be triggered when the end effector is subjected to force
Collision level	Configured via SetCollisionLevel	Configured via SetCollisionLevel, or adjusted by SetFCCollision to modify collision force and torque

Force sensor collision detection settings process:

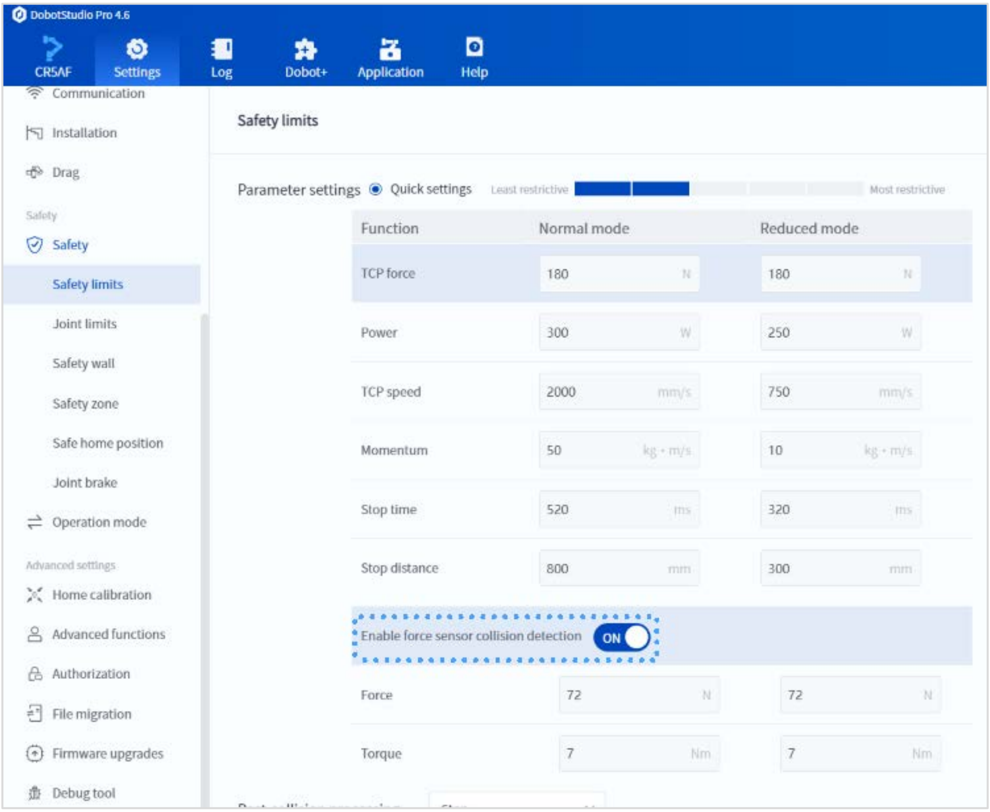
1. After connecting to the CRAF robot using DobotStudio Pro, switch the **Role** to **Administrator** (default password: 888888) on the homepage.



2. Enter the "Settings > Safety settings > Safety limits" page.



3. Click the "**Edit**" button to enable the force sensor collision detection switch. Set the collision response method and backoff distance.



NOTICE

1. Click the progress bar next to "Quick settings" to simultaneously adjust the collision level for both joint collision and force sensor collision.

2. To modify the collision force or torque for the force sensor collision detection individually, switch the Parameter settings to "**Customize**" mode.

6. Programming Commands

6.1 Blockly programming

1. 

Description: Zero the six-axis force sensor.

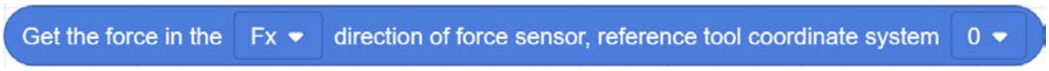
2. 

Description: Get the force/torque in the specified direction from the force sensor within the current tool coordinate system.

Parameter: Specify the direction to get the force/torque.

Return:

- When the direction is specified as Fx/Fy/Fz, it returns the force in the specified direction, unit: N.
- When the direction is specified as Mx/My/Mz, it returns the torque in the specified direction, unit: N.m.

3. 

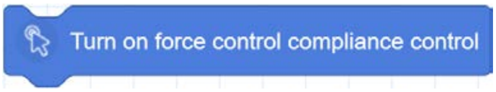
Description: Get the force/torque in the specified direction from the force sensor within the specified tool coordinate system.

Parameter:

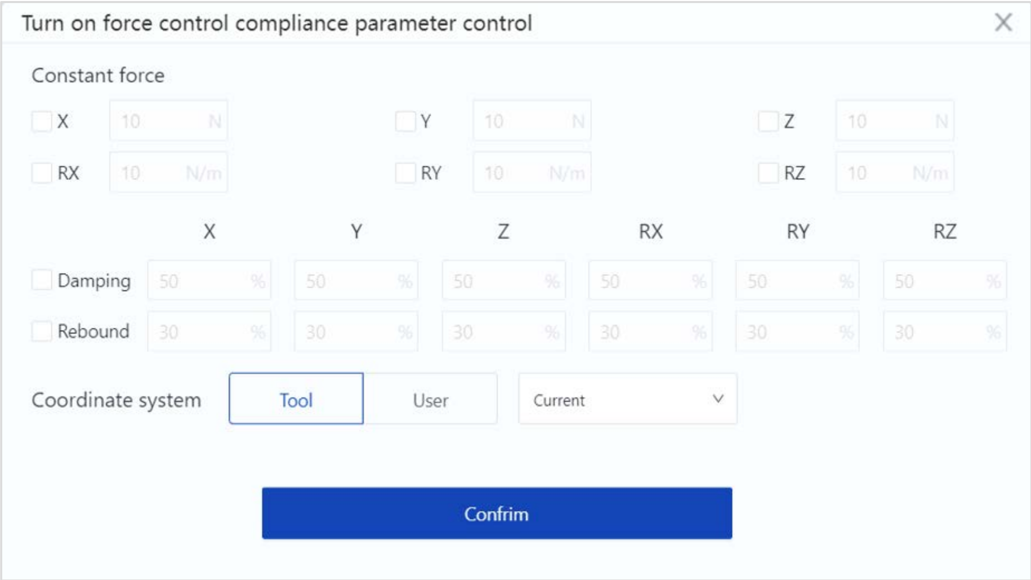
- Specify the direction to get the force/torque.
- Specify the specified tool coordinate system.

Return:

- When the direction is specified as Fx/Fy/Fz, it returns the force in the specified direction, unit: N.
- When the direction is specified as Mx/My/Mz, it returns the torque in the specified direction, unit: N.m.

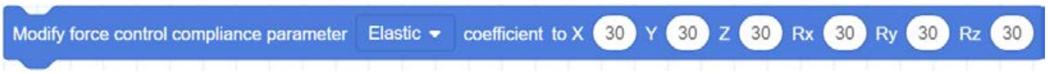
4. 

Description: Temporarily turn on force compliance control during blockly programming, and restore the original status after exiting the program.



Parameter:

- Constant force: Displacement direction X/Y/Z parameter range: [-200, 200], unit: N. Posture direction RX/RY/RZ parameter range: [-12, 12], unit: N/m.
- Damping: Range [0, 1000].
- Rebound: Range [0, 10000].
- Specify the reference tool coordinate system and user coordinate system.

5. 

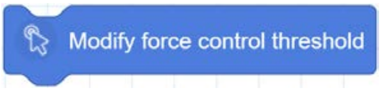
Description: Temporarily modify the parameters for force compliance during blockly programming, and restore the original parameter values after exiting the program. For detailed information about the elasticity coefficient, damping coefficient, and inertia coefficient, please refer to the appendix Impedance control.

Parameter:

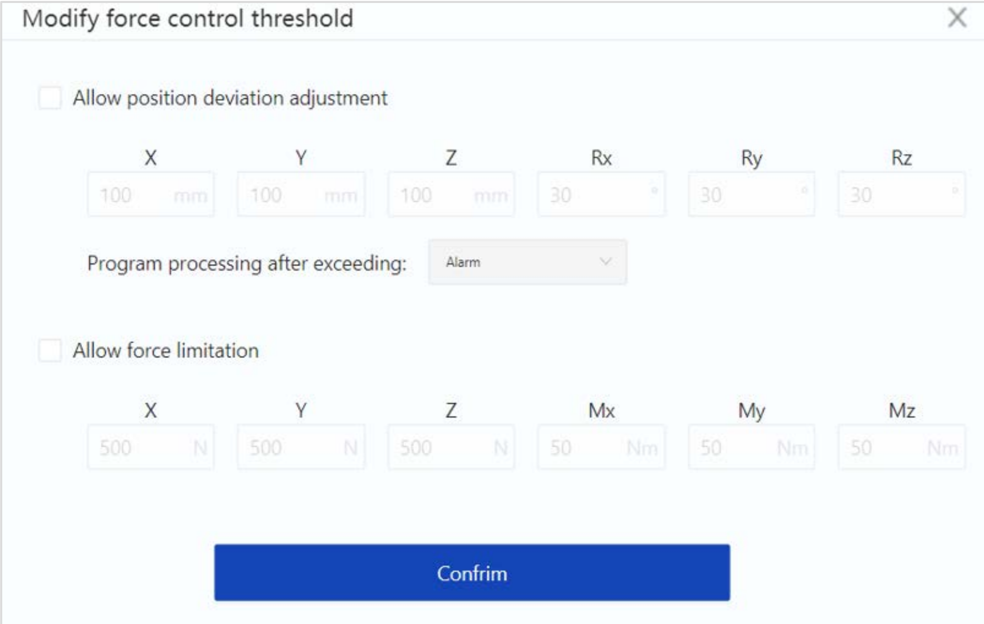
- Elastic coefficient in any direction: range (0, 10000].
- Damping coefficient in any direction: range [0, 1000].
- Inertia coefficient in any direction: range (0, 10000].

6. 

Description: Temporarily adjust the force control speed limit during blockly programming, and restore the original parameter values after exiting the program. The speed unit for the XYZ direction is mm/s, while the speed unit for the RxRyRz direction is °/s.

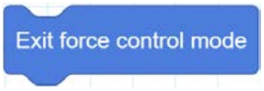
7. 

Description: Temporarily modify the force control threshold during blockly programming and restore the original parameters after exiting the program.

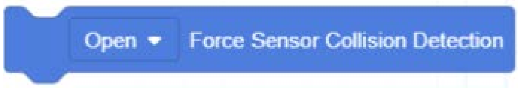


Parameter:

- Allow position deviation adjustment: X/Y/Z range (0, 1000], Rx/Ry/Rz range (0, 360].
- Program processing upon exceeding the threshold: Alarm or Continue running.
- Allow force limitation: X/Y/Z range: (0, 500]. Mx/My/Mz range (0, 50].

8. 

Description: Temporarily exit force control mode during blockly programming, and restore the original status after exiting the program.

9. 

Description: Enable/disable collision detection function of the force sensor during blockly programming, and restore to the original status after exiting the program.

10. 

Description: Temporarily modify the collision detection force and torque values of the force sensor during the blockly programming, and restore the original parameter values after exiting the program.

6.2 Script programming

1. SixForceHome

Command:

```
SixForceHome()
```

Description: Zero the force sensor.

Return: The zeroing result. 0: Zeroing successful. Not 0: Zeroing failed.

Example:

```
local res = SixForceHome()
```

2. GetForce

Command:

```
GetForce(tool)
```

Description: Get the force and torque applied in the TCP direction.

Optional parameter:

"tool": Select the calibrated tool coordinate system, range: [0, 50]. If not specified, it defaults to the global tool coordinate system.

Return:

{Fx, Fy, Fz, Mx, My, Mz}. Where, "Fx, Fy, Fz" are the force values in the X, Y, Z directions (units: N), "Mx, My, Mz" are the torque values in the RX, RY, RZ directions (units: N/m).

Example:

```
local force = GetForce(1)
Print(force)
```

3. FCForceMode

Command:

```
direction={1,1,1,1,1,1}
force={10,10,10,10,10,10}
FCForceMode(direction,force,{reference = 0,user=0,tool=0})
```

Description: Temporarily enable force control with user-specified parameters during script programming, and restore the original status after exiting the script.

Required parameter:

- "direction": Enable/disable force control adjustment for a specific direction.

  1: Enable. 0: Disable.

- "force": The target force for a specific direction. Displacement direction range: [-200, 200], unit: N. Posture direction range: [-12, 12], unit: N/m. When the target force is set to 0, it enters compliance mode, which is similar to force-control drag. If force control is not enabled for a specific direction, the target force for that direction will not take effect.

Optional parameter:

- "reference": The reference coordinate system.

  reference=0: Tool coordinate system. reference=1: User coordinate system.
- "user": Select the calibrated user coordinate system, range:[0, 50].
- "tool": Select the calibrated tool coordinate system, range: [0, 50].

Example:

```
FCForceMode({1,1,1,1,1,1},{10,10,10,10,10,10},{reference=0,tool=1})
```

4. FCSetStiffness

Command:

```
FCSetStiffness({x,y,z,rx,ry,rz})
```

Description: Temporarily modify the elasticity coefficients in each direction under force control mode during script programming, and restore the original parameter values after exiting the script. For detailed information about the elasticity coefficients, please refer to the Impedance control in the appendix.

Required parameter:

{x, y, z, rx, ry, rz}: range (0, 10000], default is 30.

Example:

```
FCSetStiffness({30,30,30,30,30,30})
```

5. FCSetDamping

Command:

```
FCSetDamping({x,y,z,rx,ry,rz})
```

Description: Temporarily modify the damping coefficients in each direction under force control mode during script programming, and restore the original parameter values after exiting the script. For detailed information about the damping coefficients, please refer to the Impedance control in the appendix.

Required parameter:

{x, y, z, rx, ry, rz}: range [0, 0, 1000], default is 50.

Example:

```
FCSetDamping({50,50,50,50,50,50})
```

6. FCSetMass

Command:

```
FCSetMass({x,y,z,rx,ry,rz})
```

Description: Temporarily modify the inertia coefficients in each direction under force control mode during script programming, and restore the original parameter values after exiting the script. For detailed information about the inertia coefficients, please refer to the Impedance control in the appendix.

Required parameter:

{x, y, z, rx, ry, rz}: range (0, 0, 1000], default is 20.

Example:

```
FCSetMass({20,20,20,20,20,20})
```

7. FCSetDeviation

Command:

```
FCSetDeviation({x,y,z,rx,ry,rz},controltype)
```

Description: Temporarily modify the displacement and posture deviation under force control mode during script programming, and restore the original parameter values after exiting the script. If the constant force deviates significantly during the force control process, the robot will respond accordingly.

Required parameter:

- "x, y, z": Displacement deviation in force control mode. Range: (0, 1000], unit: mm; default is 100 mm.
- "rx, ry, rz": Posture deviation in force control mode. Range: (0, 360], unit: °; default is 36°.

Optional parameter:

"controltype": The handling method when the threshold is exceeded during force control.

- 0: When the threshold is exceeded, the robot will trigger an alarm.
- 1: When the threshold is exceeded, the robot will stop searching and continue running along the original trajectory.

Example:

```
FCSetDeviation({100,100,100,30,30,30}, 0)
```

8. FCSetForceLimit

Command:

```
FCSetForceLimit({x,y,z,rx,ry,rz})
```

Description: Temporarily modify the maximum force limit for each direction (affects all directions, including those not enabled for force control) during script programming, and restore the original parameter values after exiting the script.

Required parameter:

- "x, y, z": range (0, 500], default is 500.
- "rx, ry, rz": range (0, 50], default is 50.

Example:

```
FCSetForceLimit({500,500,500,20,20,20})
```

9. FCSetForceSpeedLimit

Command:

```
FCSetForceSpeedLimit({x,y,z,rx,ry,rz})
```

Description: Temporarily modify the force control speed limit for each direction (affects all directions, including those not enabled for force control) during script programming, and restore the original parameter values after exiting the script. When the force control speed limit is set lower, the adjustment speed is slower, making it suitable for gentle, low-speed contact. When the force control speed limit is set higher, the adjustment speed is faster, making it suitable for high-speed force control applications. It should be adjusted according to the specific application requirements.

Required parameter:

- "x,y,z": Force control speed limits for displacement directions. For CRAF models, range: (0, Safety limit TCP speed].
- "rx,ry,rz": Force control speed limits for posture directions. For CRAF models, range: (0, (4 x Safety limit TCP speed x 0.001 / 3.14 × 180)].

Example:

```
FCSetForceSpeedLimit({20,20,20,20,20,20})
```

10. FCOff

Command:

```
FCOff()
```

Description: Temporarily exit force control mode during script programming, and restore the original status after exiting the script.

Example:

```
FCOff()
```

11. FCCollisionSwitch

Command:

```
FCCollisionSwitch(switch)
```

Description: Temporarily enable/disable force sensor collision detection switch during script programming, and restore the original status after exiting the script.

Required parameter:

When the switch is set to 0, the force sensor collision detection function is disabled; when the switch is set to 1, the function is enabled.

Example:

```
FCCollisionSwitch(1)
```

12. SetFCCollision

Command:

```
SetFCCollision(force,torque)
```

Description: Temporarily modify the collision detection threshold parameters of the force sensor during script programming, and restore the original status after exiting the script.

Required parameter:

- "force": The force threshold for triggering the collision detection of the force sensor, unit: N,　range: (0,300].
- "torque": The torque threshold for triggering force sensor collision detection, unit: Nm, range: [0, 20].

Example:

```
SetFCCollision(10,2)
```

NOTICE

Non-CRAF models will report errors when calling the FCCollisionSwitch or SetFCCollision commands.

6.3 Script Demo

```
MovJ(P1) -- Move to the waiting position P1
SixForceHome() -- Zero the force sensor.
DO(2,1)
VelL(2) -- Set speed
MovL(P2) – Move to the point P2 that fits the contact surface
direction={0,0,1,0,0,0} -- Set force control direction Fz
force={0,0,4,0,0,0} -- Apply a 4N downward force in the Fz direction

FCForceMode(direction,force,{reference = 0,user=0,tool=0}) -- Enable force control
FCSetDamping({50,50,50,50,50,50}) -- Set the damping coefficients
FCSetStiffness({30,30,30,30,30,30}) -- Set the stiffness coefficients
FCSetMass({20,20,20,20,20,20}) -- Set the inertia coefficients
FCSetForceSpeedLimit({20,20,20,20,20,20}) -- Set the maximum speed for force-control motion
MovL(P3) – Move to point P3 with a force of 4N (with force control effect)
MovL(P4) -- Move to point P4 with a force of 4N (with force control effect)
FCSetForce({0,0,8,0,0}) -- Adjust the contact force to 8N
MovL(P5) – Move to P5 with a force of 8N (with force control effect)
FCOff() -- Disable force control
VelL(50)
MovL(P6)
DO(1,0)
MovL(P1)
```


6.4 TCP/IP secondary development

The TCP/IP commands related to force control are shown in the following table. For detailed information about commands, please refer to *Dobot TCP_IP Remote Control Interface Guide V4.6.0*.

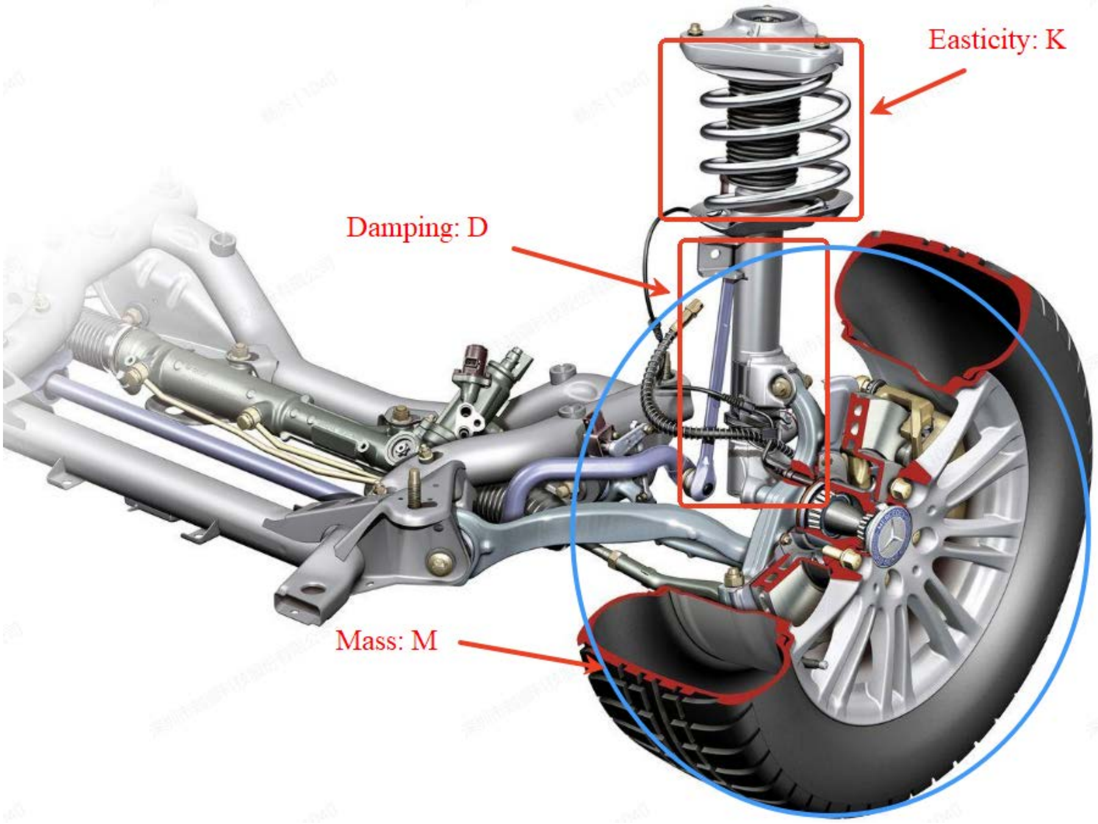
Command	Function	Command type
EnableFTSensor	Switch on/off the force sensor	Immediate command
SixForceHome	Zero the force sensor	Immediate command
GetForce	Get force sensor data	Immediate command
ForceDriveMode	Enter the force-control drag mode	Immediate command
ForceDriveSpeed	Set the force-control drag speed	Immediate command
FCForceMode	Enable force control with user-specified parameters	Queue command
FCSetDeviation	Set the offset and posture deviation in force control mode	Immediate command
FCSetForceLimit	Set the maximum force limit	Immediate command
FCSetMass	Set the mass coefficients for each direction in force control mode	Immediate command
FCSetStiffness	Set the stiffness coefficients for each direction in force control mode	Immediate command
FCSetDamping	Set the damping coefficients for each direction in force control mode	Immediate command
FCOff	Exit the force control mode	Queue command
FCSetForceSpeedLimit	Set the speed limits for force control adjustment in each direction	Immediate command
FCSetForce	Adjust the constant force settings in real time	Immediate command
SetFCCollision	Set the threshold parameters for force sensor collision detection (Applicable to CRAF model only)	Immediate command
FCCollisionSwitch	Switch on/off the force sensor collision detection (Applicable to CRAF model only)	Immediate command

Appendix A Impedance Control

In practical force control applications, the adjustment of three key impedance control parameters (stiffness coefficient, damping coefficient, and inertia coefficient) is often involved. These parameters critically influence the robot's force control performance. By dynamically adjusting appropriate parameter values, the robot can "sense" external forces and automatically adapt its trajectory, achieving active compliance to suit applications such as precision assembly, polishing, grinding, and robot massage.

A.1 Get to know impedance control

Here takes the frequency selective damping in a car as an example to introduce impedance control.



The frequency selective damping (FSD) in a car can only adjust the damping D, as both the spring stiffness K and the wheel mass M are fixed.

Impedance control uses software to simulate a "mechanical damper", enabling simultaneous adjustment of mass M (inertia), easticity K (stiffness), and damping D to balance the comfort and stability of the vehicle during driving.

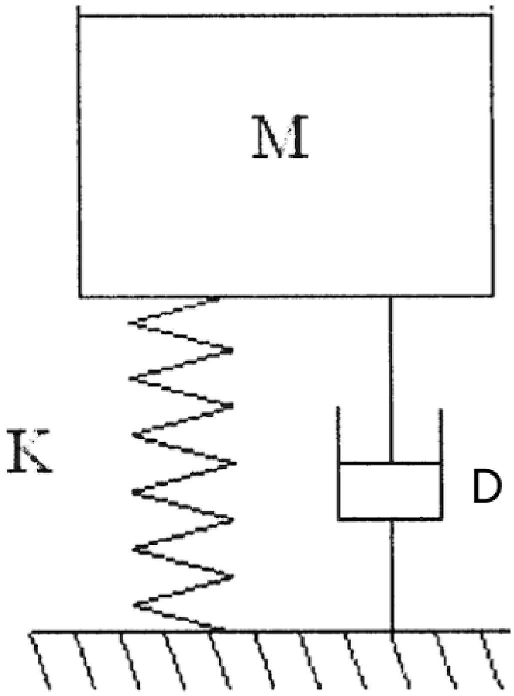
- **Inertia (mass):** Equivalent to the "response speed" of a wheel. Increasing the mass parameter slows the wheel's response to bumps, making it better suited for smooth, high-speed roads; decreasing the mass parameter makes the wheel more responsive, ideal for off-road terrain.
- **Elasticity (stiffness):** Equivalent to the "softness or hardness" of a spring. Increasing stiffness reduces body roll but results in a harsher ride (e.g., track mode); decreasing stiffness improves bump absorption (e.g., comfort mode).
- **Damping (resistance):** Equivalent to the "firmness" of a damper. Increasing damping quickly suppresses body oscillations (e.g. when crossing speed

bumps); reducing damping allows controlled residual motion (e.g. for long-wave undulating roads).

Adjust the inertia, elasticity and damping parameters according to the following different scenarios:

- **High-speed driving (stability prioritized):** Increase elasticity (to reduce roll) and enhance damping (to suppress vibrations).
- **Bumpy roads (comfort prioritized):** Decrease elasticity (to absorb shocks) and moderately reduce damping (to allow free suspension movement).
- **Cornering handling (balancing roll and grip):** Dynamically adjust mass, reduce the inertia of outer wheels, and improve responsiveness.

A.2 Robot impedance control



- **Inertia (M):** Adjust the robot's inertia M to convert external forces/torques into velocity proportionally. High inertia suits scenarios with minimal force variations, reducing sensitivity to enhance stability and precision. Low inertia is ideal for dynamic force conditions, improving sensitivity and realizing fast response.
- **Damping (D):** Adjust the robot's damping D enables adaptive responses to force/torque fluctuations. High damping slows down the response speed of the robot and improves the stability and accuracy. Low damping realizes fast response.
- **Stiffness (K):** Adjust the robot's stiffness K tailors its compliance to different forces/torques. High stiffness ensures precision in rigid environments. Low stiffness enhances safety for soft environments or human collaboration.

A.3 ForceControl application

A.3.1 ForceControl grinding

Force control cannot handle "rigid-on-rigid" contact scenarios. Force-control grinding is like "a smooth highway": direct rigid contact without any cushioning medium can easily cause oscillation and bouncing. Therefore, grinding heads usually incorporate a sponge interlayer. By adjusting impedance parameters, constant-force grinding can be achieved.

1. Determining the starting point of force control: jog the robot to press against the grinding surface until the desired force (e.g., -10N) is reached, then set this position as the starting point of force control.
2. The grinding head should always remain perpendicular to the surface. Typically, only Z-direction force control is enabled (force control in (RX/RY directions may be considered in special cases).

> **NOTE**
>
> When grinding surfaces with higher flatness, the recommended parameter values are: inertia coefficient = 150, damping coefficient = 800, elasticity coefficient = 1000.

A.3.2 ForceControl massage

Force-control massage is similar to driving on a "bumpy road", requiring rapid response in Z direction to achieve better fitting effect. At this point, the smaller the inertia coefficient, the faster the response speed; but if the inertia is too small, oscillation can easily occur. It is recommended to adjust the inertia coefficient incrementally from small to large values to achieve both a fast regulation speed and avoid oscillation. **Reference range for the inertia coefficient: 20 – 60.**

Since the force control direction requires a fast response, damping will create resistance to the adjustment direction of the response, so the damping should not be too high. However, if the damping is too low, stability will significantly decrease. **Reference range for the damping coefficient: 50 – 200.**

The stiffness coefficient can either be left unset or assigned a relatively small value. **Reference range for the stiffness coefficient: 0 – 300.**

