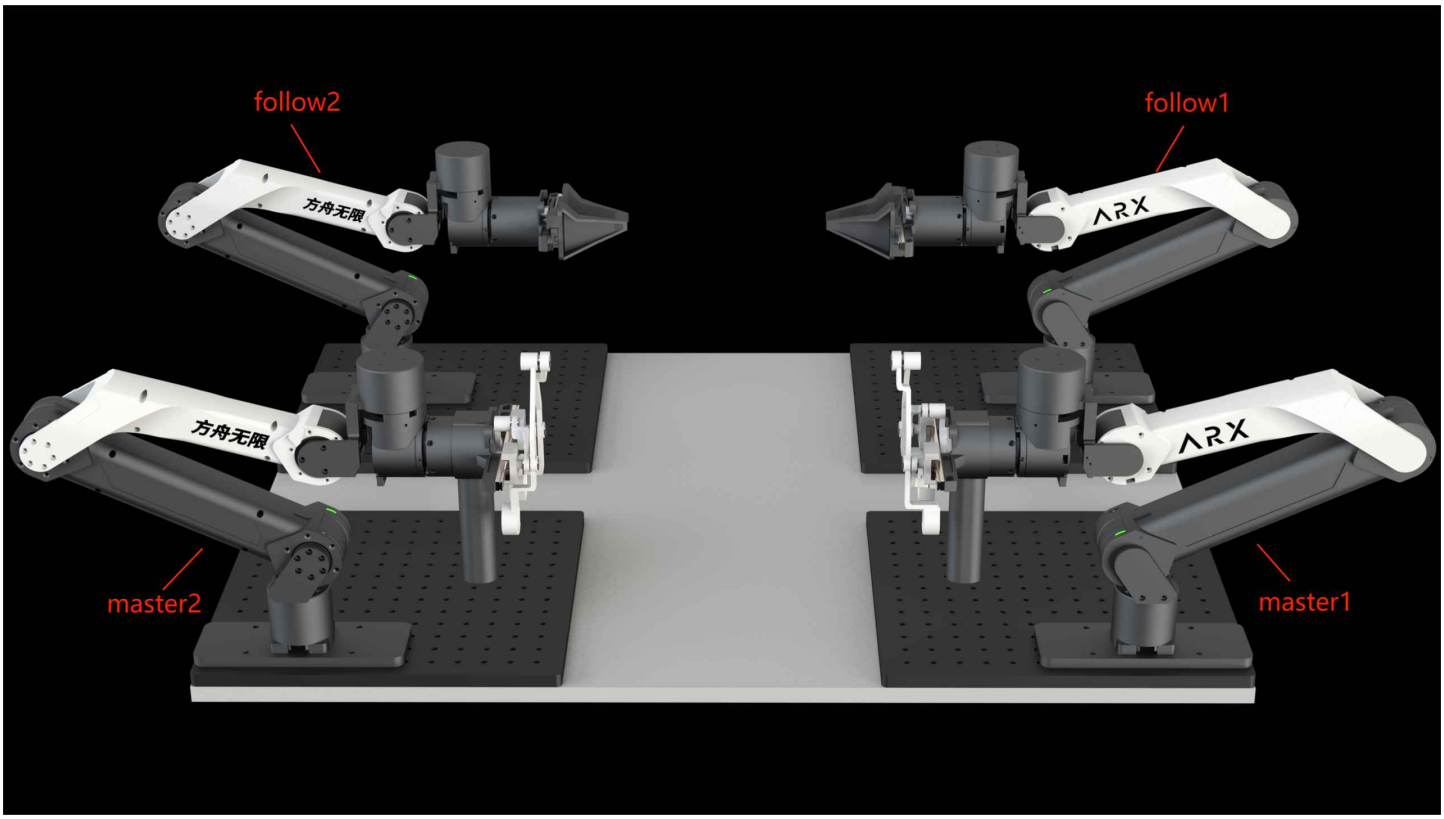
X5 Data Collection System User Manual EN



Product Introduction

Welcome to the ARX-5 ultra-light force-controlled robotic arm. This arm distinguishes from all other industrial collaboration arms from the perspective of robotics. It emphasizes the importance of lightweight, high flexibility, and complete joint force control. It can be adapted to various research purposes, including most quadruped robots, large models, embodied AI, algorithm verification, etc.

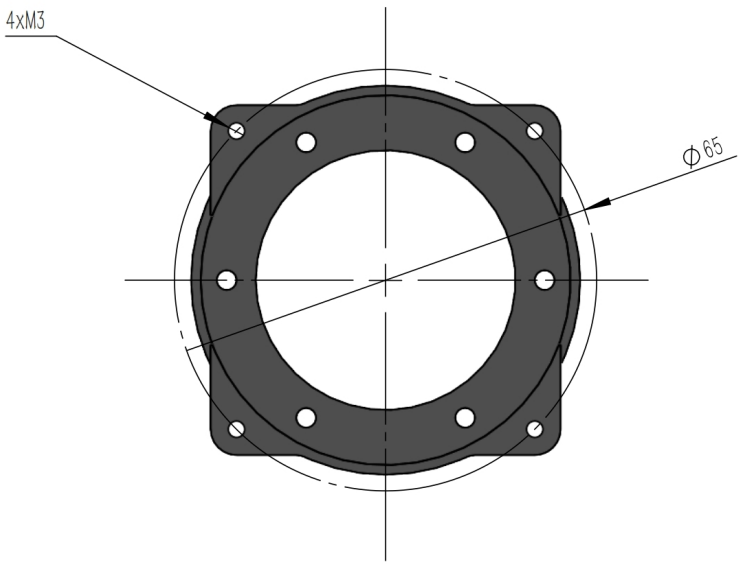
Packing List

Robotic Arm	**1 pc**
Ecat to can	**1 pc**
M3 X 10 Screw	**4 pcs**
DC24V Power Supply	**1 pc**
AC Power Cord	**1 pc**
M3 Wrench	**1 pc**
G-shaped Clamp	**2 pcs**
Table Top Mounting Plate	**1 set**

Set Up

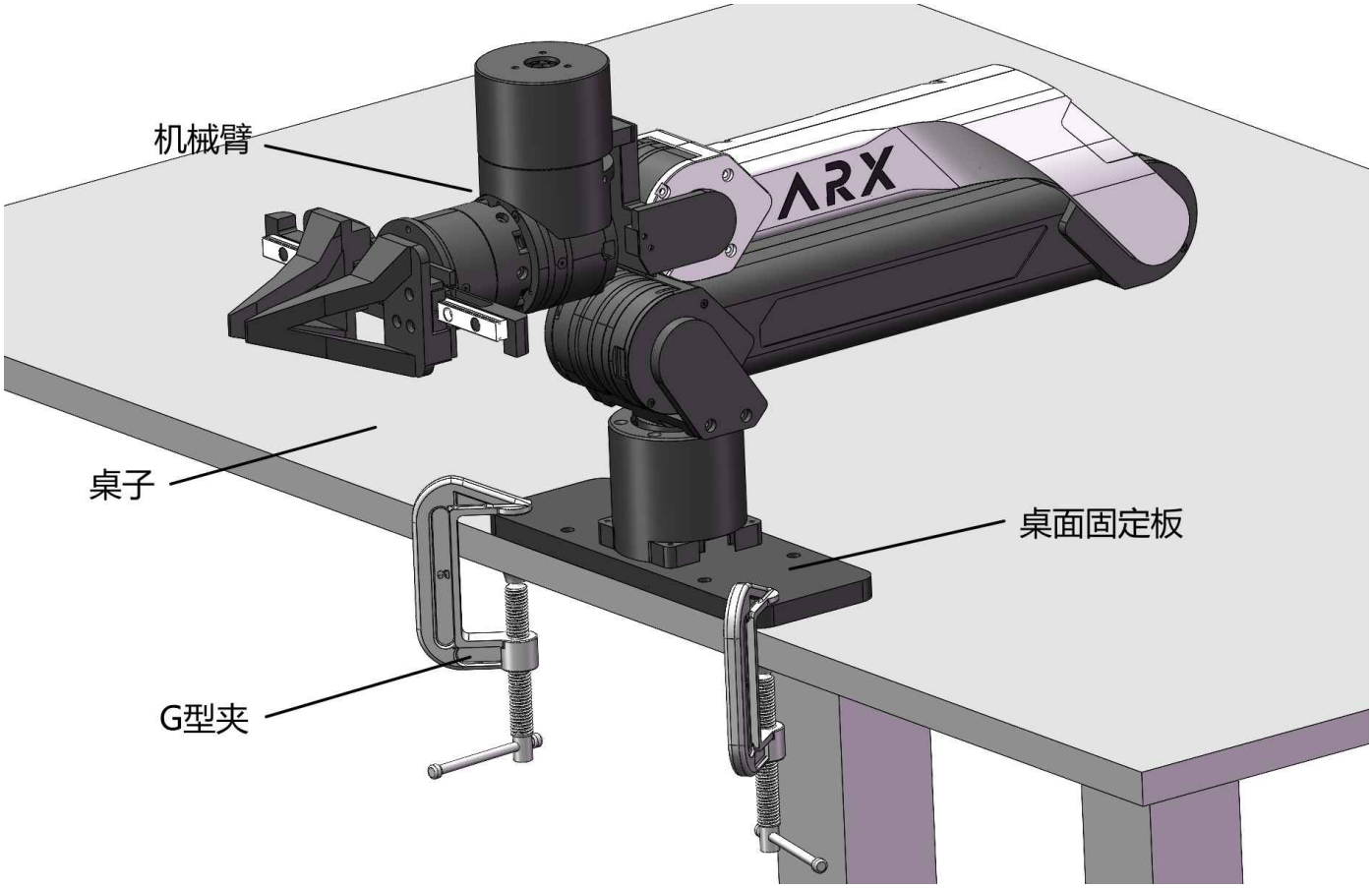
Fixating Robotic Arm

The four M3 screw holes in the bottom of the arm can be used for making a base that fits the robot.



机械臂底座螺丝安装示意图

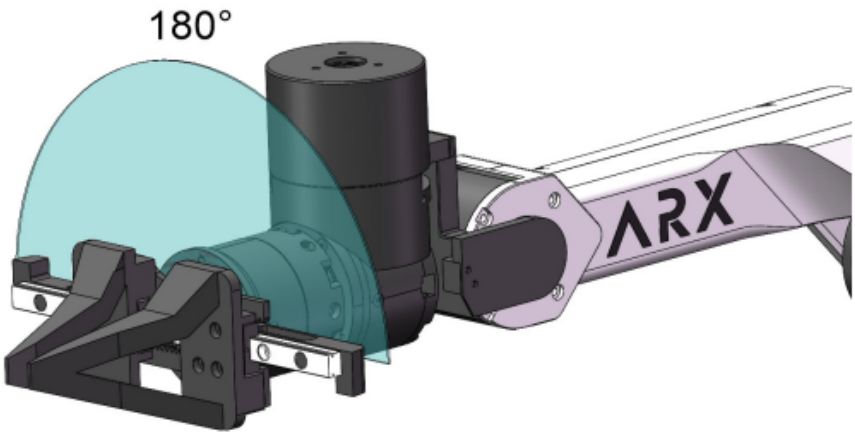
To fix the arm on table top, two G-clips and the Table Top Mounting Plate needs to be set up as shown below.

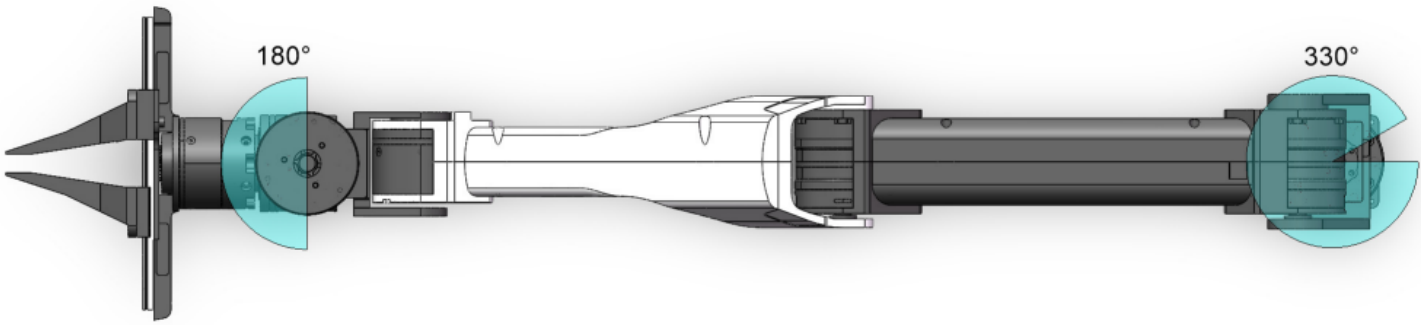


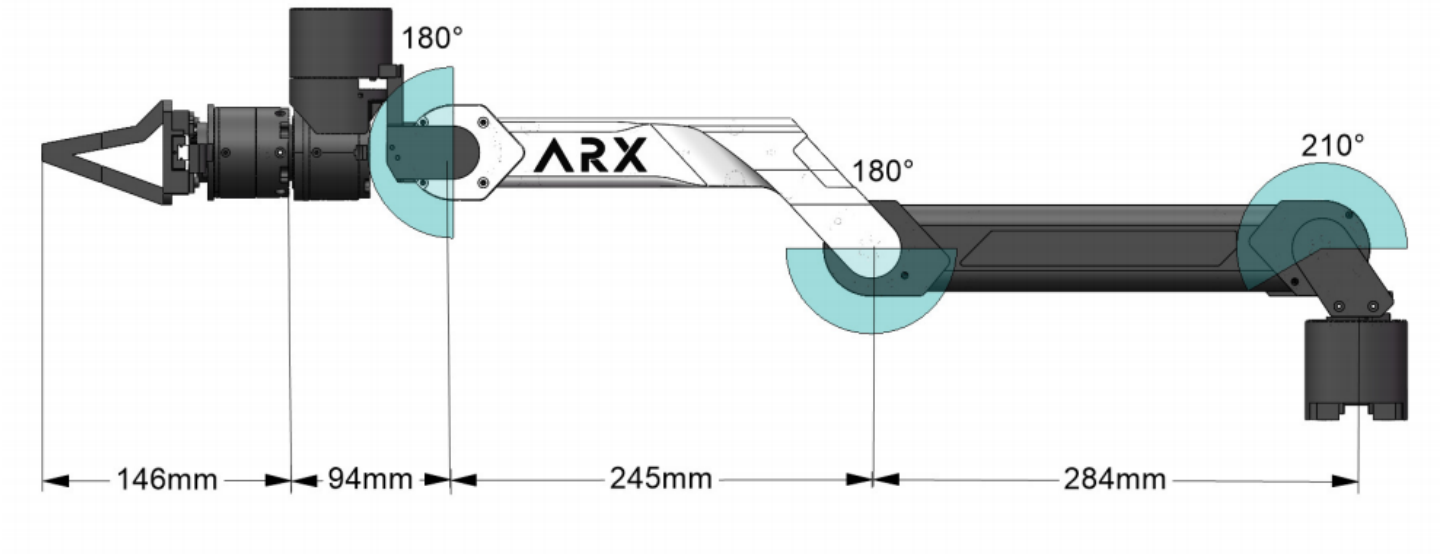
When looking from above with the gripper facing forward, the cable is on its left.

Product Parameters

自由度	6+gripper
自重	≈2.95kg
带夹爪总重	≈3.35kg
额定负载	1.5kg
峰值负载	3kg
臂展	620mm
电源需求	24V 10A-20A
通信接口	CAN
控制类型	位置+力控
控制系统	Ubuntu
功率	峰值500W
力反馈与碰撞检测	有

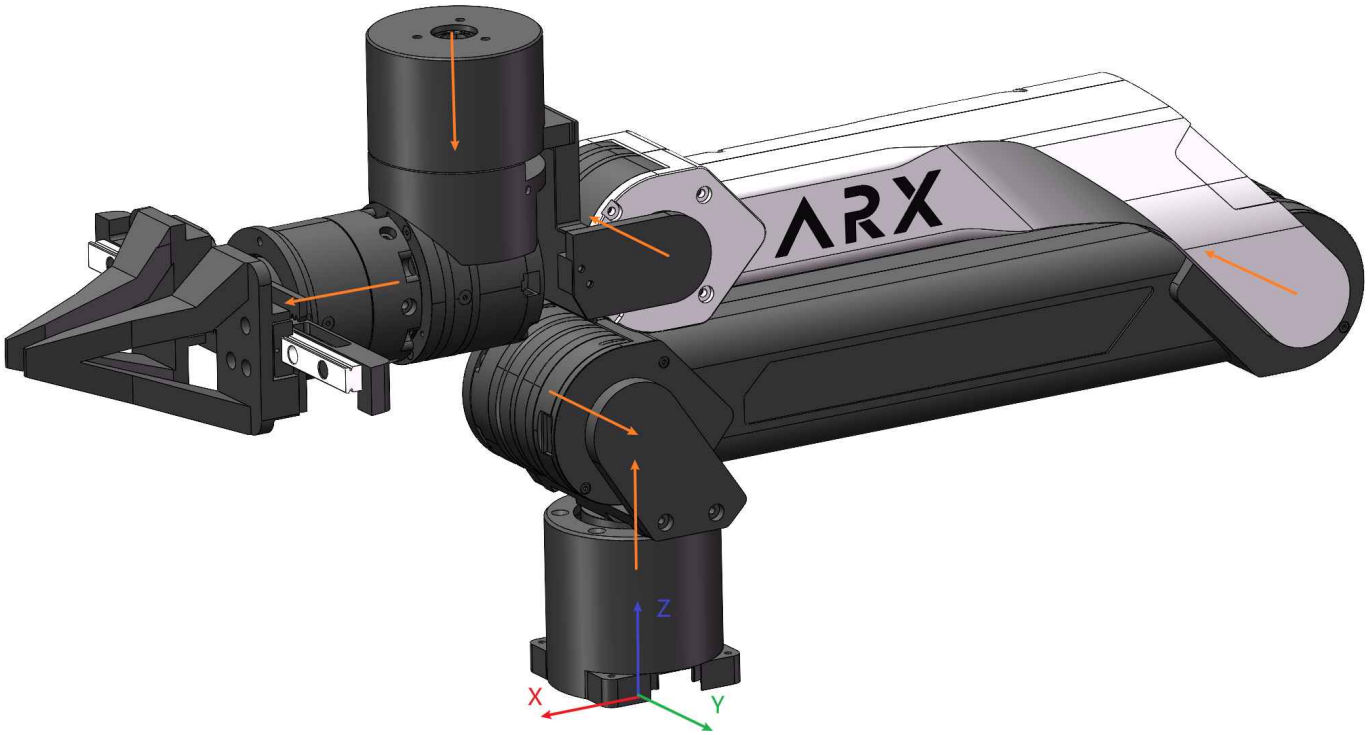






Rotation Direction and Coordinate System

Based on the right-hand rule



Use of Robot Arm

Dependency Installation and Environment Installation

Please make sure to follow this installation order.

ROS Installation

Install Ubuntu system 20.04.

Install ROS1, we recommend installation with fishros with the following command. Currently, it only supports ROS1.

```
wget http://fishros.com/install -O fishros && . fishros
```

Configure CAN Environment

```
Configure can
sudo apt install can-utils
sudo apt install net-tools
```

Keyboard Detection

```
sudo apt-get install libevdev-dev
```

Install the KDL library:

```
Choose a library to save path, and execute
git clone https://github.com/orocos/orocos_kinematics_dynamics.git
Enter orocos_kdl directory
mkdir build
cd build
cmake ..
make
sudo make install
Complete Installation


Choose a library to save path, and execute
git clone https://github.com/ros/kdl_parser.git
Enter kdl_parser directory:
mkdir build
cd build
cmake ..
make
After compilation complete
sudo make install
Complete Installation
```

Hardware Connection

Plug in Type-c and Ethernet

Ethernet can be plugged into either port

Type-c is only used for power supply

Use one adapter board for each robotic arm

Check Ethernet ID

Please note plug in and read the ID one by one

```
1 ifconfig -a
```

Find Ethernet Connection

```
enx207bd2d3159b: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet6 fe80::25d1:8450:e282:cbdf  prefixlen 64  scopeid 0x20<link>
        ether 20:7b:d2:d3:15:9b  txqueuelen 1000  (以太网)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 74  bytes 12551 (12.5 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

In this example, enx207bd2d3159b is the ethernet ID

Depending on the equipment (for example, for two leaders and two followers, read 4 times, record the ID individually)

Change Program Ethernet ID

Enter master1, master2, follow1, and follow2 separately

Change the Ethernet ID in arx.cpp to the corresponding ID

```
char phy[] = "enx00e04c36119f";
```

Compile and Run

Change Ethernet ID and save

Execute make.sh

```
./make.sh
```

After compilation, execute the remote.sh script under remote_control to start running

```
sudo ./remote.sh
```

Exception Handling

Unable to Control

1. USB port might have been loose. Restart the program.

Robotic Arm Jittering

This robotic arm uses planetary reducer gearbox, therefore backlash exists. When the load varies significantly, and the control mode is not optimized enough, it may cause the arm to jitter.

Unable to Connect with CAN

1. Check whether the base port is plugged in securely

2. Note the installation order of the environment. Install ROS first

3. Check whether the USB2CAN hardware port is damaged (type-c port and gh1.25 port)

4. There may have been a possible disconnection program protection situation. The cable will need to be re-plugged, and the arm needs to reboot.

Disabled during Operation

1. Power supply needs to be 360W and above

2. To protect the motor, when the arm collides into objects, it may trigger stall protection. The program needs to be reexecuted.

Emergency Handling

1. The arm comes with self protection program. If a collision occurs, the damping mode will be triggered. At this point, all terminals need to be closed off, and the power should to be turned off when necessary

2. If a singular solution occurs, press R to reset

3. Make sure there are no valuable items around the arm before switching it on to prevent damage

4. When using the arm for the first time, pay attention to the orientation of the robotic arm. Look down at the robotic arm with the gripper facing forward, the cable on the bottom should be on the left to prevent collision with other items when it returns to zero position.