

ROS2

```
# Shut off auto-start
sudo systemctl restart start_app_node.service
# Start auto-start
sudo systemctl restart start_app_node.service
# Shut down all ROS processes ROS
~/stop_ros.sh

# Handle control
ros2 launch sdk jetarm_sdk.launch.py
ros2 launch peripherals joystick.launch.py

#####Basic control of the robotic arm. #####
# Control the servo
#1. Control the steering gear through topics.
ros2 launch sdk jetarm_sdk.launch.py
ros2 topic list
ros2 topic pub /ros_robot_controller/bus_servo/set_position ros_robot_controller_msgs/msg/ServosPosition '{"position": [{"id": 1, "position": 700}]}'
#2. Use python file control. python
ros2 launch sdk jetarm_sdk.launch.py
python3 ~/ros2_ws/src/example/example/simple/include/bus_servo_node.py
#3. Launch file startup control launch
ros2 launch example bus_servo.launch.py

# Control LED LED
#1. Control the LED through topics. LED
ros2 launch sdk jetarm_sdk.launch.py
ros2 topic list
ros2 topic pub /ros_robot_controller/set_led ros_robot_controller_msgs/msg/LedState "{id: 1,on_time: 0.1,off_time: 0.2,repeat: 10}"
#2 Use python file control. python
ros2 launch sdk jetarm_sdk.launch.py
python3 ~/ros2_ws/src/example/example/simple/include/led_node.py
#3 Launch file startup control launch
ros2 launch example led.launch.py

# Buzzer Control
#1. Buzzer control node
ros2 launch sdk jetarm_sdk.launch.py
ros2 topic list
ros2 topic pub /ros_robot_controller/set_buzzer ros_robot_controller_msgs/msg/BuzzerState "{freq: 1900, on_time: 0.1,off_time: 0.2,repeat: 10}"
#2. Use python file control. python
ros2 launch sdk jetarm_sdk.launch.py
python3 ~/ros2_ws/src/example/example/simple/include/buzzer_node.py
#3. Launch file startup control launch
ros2 launch example buzzer.launch.py

# Robotic Arm PC
cd arm_pc
python3 main.py

# Servo debug tool
cd factory_utils/Bus_Servo_Tool
python3 main.py

#####Advanced control of the robotic arm. #####
# Forward Kinematics
#1. Invocation through service
ros2 launch sdk jetarm_sdk.launch.py
ros2 service list
ros2 service call /kinematics/set_joint_value_target kinematics_msgs/srv/SetJointValue "{joint_value: [500.0, 400.0, 300.0, 400.0, 500.0]}"
#2. Launch file startup control launch
ros2 launch example fk.launch.py

# Inverse kinematics
#1. Invocation through service
ros2 launch sdk jetarm_sdk.launch.py
ros2 service list
ros2 service call /kinematics/set_pose_target kinematics_msgs/srv/SetRobotPose "{position: [0.35, 0.0, 0.24], pitch_range: [-180,180], pitch: 10, resolution: 1}"
#2. Launch file startup control launch
ros2 launch example ik.launch.py

#####ROS+OpenCV Courses. ROS+OpenCV#####
```

```
# Camera invocation
ros2 launch peripherals depth_camera.launch.py

# Color threshold
ros2 launch example color_threshold.launch.py

# Color space conversion
ros2 launch example color_space.launch.py

# Color recognition
ros2 launch example color_recognition.launch.py

# Pixel coordinate calculation
ros2 launch example pixel_coordinate_calculation.launch.py

# Object pose calculation
ros2 launch example object_attitude_calculation.launch.py

# Coordinate system transformation
ros2 launch example coordinate_system_transformation.launch.py

# Trajectory planning
ros2 launch example path_planning.launch.py

# Positioning and clamping
ros2 launch example positioning_clamp.launch.py

# Color sorting
ros2 launch example color_sorting_node.launch.py

# Color block tracking
ros2 launch example color_tracking.launch.py

# Tag tracking
ros2 launch example tag_track_node.launch.py

# KCF item tracking KCF
ros2 launch example kcf_track_node.launch.py

#####Deep learning applications. #####
#Waste classification by robotic arm
ros2 launch example waste_classification.launch.py

# Three-dimensional face detection
ros2 launch example face_mesh.launch.py

# Mediapipe Face Tracking Mediapipe
ros2 launch example face_tracking.launch.py

# Mediapipe hand gesture interaction Mediapipe
ros2 launch example hand_gesture.launch.py

# Mediapipe finger trajectory Mediapipe
ros2 launch example finger_trajectory.launch.py

#####Application Course of Robotic Arm Depth Camera. #####
# Pseudo-color processing of depth maps
ros2 launch example get_depth_rgb_img.launch.py

# Distance measurement
ros2 launch example distance_measure.launch.py

# Conversion of depth maps
ros2 launch example rgb_depth_to_pointcloud.launch.py

# Height detection clamping
ros2 launch example remove_too_high.launch.py

# Three-dimensional space grasping
ros2 launch example track_and_grab.launch.py

# Three-dimensional shape sorting
ros2 launch example shape_recognition.launch.py

#####Voice control #####
# Start the microphone node
#xf
```

```
ros2 launch xf_mic_asr_offline mic_init.launch.py enable_setting:=true

#wonderechopro
ros2 launch werechopro wonderechopro_init.launch.py

# Voice-controlled color sorting
ros2 launch xf_mic_asr_offline voice_control_color_sorting.launch.py

# Voice-controlled color tracking
ros2 launch xf_mic_asr_offline voice_control_color_track.launch.py

#####Large model #####
#####online #####
# Invocation of large language models
python3 large_models/llm_demo.py

# Semantic understanding
python3 large_models/llm_nlu_demo.py

# Emotional perception
python3 large_models/llm_er_demo.py

# Voice wake-up
python3 large_models/wakeup_demo.py

# Speech recognition
python3 large_models/asr_demo.py

# Speech synthesis
python3 large_models/tts_demo.py

# Voice interaction
python3 large_models/voice_interaction_demo.py

# Vision language model invocation
python3 large_models/vllm_understand.py

# Object Recognition in the application of Vision language Models
PYTHON3 large_models/vllm_detect_demo.py

# Scene Understanding in the Application of Vision language Models
Python3 large_models/vllm_understand.py

# Text recognition in the application of Vision language Models
python3 large_models/python3 vllm_ocr.py

# Voice control of multimodal models
ros2 launch large_models_examples llm_control_servo.launch.py

# Color block Sorting of multimodal models
ros2 launch large_models_examples llm_object_sorting.launch.py

# Waste classification of multimodal models
ros2 launch large_models_examples llm_waste_classification.launch.py

# Three-dimensional object sorting of multimodal models
ros2 launch large_models_examples llm_3d_object_sorting.launch.py

# Real-time detection of embodied intelligent applications
ros2 launch large_models_examples vllm_with_camera.launch.py

# Visual tracking for embodied intelligence applications
ros2 launch large_models_examples vllm_track.launch.py

# Intelligent handling for embodied intelligence applications
ros2 launch large_models_examples vllm_object_transport.launch.py

# Intelligent smart home assistant for embodied intelligent applications
ros2 launch large_models_examples vllm_dietitian.launch.py

#####JetArm robotic arm + slide rail Combined course JetArm+#####
# Color block sorting
ros2 launch stepper color_sorting.launch.py

# Waste classification
ros2 launch stepper waste_classification.launch.py

# Tag sorting
ros2 launch stepper apriltag_sorting_stepper.launch
```

```
#####JetArm+ microwheel chassis control JetArm+#####
```

```
# Target tracking
```

```
ros2 launch chassis object_tracking.launch.py
```

```
# Click on the target that needs to be tracked and enter the instruction to start the tracking service. ,
```

```
ros2 service call /object_tracking/set_running std_srvs/srv/SetBool "{data: true}"
```

```
# Stop the feature
```

```
ros2 service call /object_tracking/set_running std_srvs/srv/SetBool "{data: false}"
```

```
# Line following driving
```

```
ros2 launch chassis line_following_node.launch.py
```

```
# Click on the line following target and enter the instruction to start the line following service. ,
```

```
ros2 service call /line_following/set_running std_srvs/srv/SetBool "{data: true}"
```

```
# Stop the feature
```

```
ros2 service call /line_following/set_running std_srvs/srv/SetBool "{data: false}"
```

```
#####JetArm+#####
```

```
# Color block sorting
```

```
ros2 launch motor color_sorting.launch.py
```

```
# After the gameplay is launched, click on the camera screen window and press 'a' to start sorting. Press 's' to stop sorting. 'a's'
```