JAKA

Just Always Keep Amazing

SDK V2.2.2 Release Notes

Contents

1. Release Information

1.1 Release Date

2024/11/20

1.2 Version Information

This SDK version 2.2.2 is compatible with Linux x86_64 bit and Windows x86_64 systems.

Other detailed information is as follows:

Name	Version	Note
Controller	1.7.1_45 and above	X64 / X32
SCB	03_11_R	/
PSCB	03_13_PR	/
Zu, C, PRO series servo	R3200	/
MiniCobo series servo	R2200_MINICOBO	/

Note:

The compatibility between SDK V2.2.2 and the controller version should be given attention.

- SDK V2.2.2 should be used on controller of version 1.7.1_45 and above.
- For controllers of version 1.7.0_x and 1.5.x, it is recommended to use SDK V2.1.11 and its previous versions.
- If the controller's version and the SDK's version mismatch, some interfaces (interfaces to get status like get_tcp_position(); get_joint_position(); is_in_collision(); is_in_pos(); interfaces to block joint move and linear movement) would be unable to be used, and some errors would occur.
- The reason is that interfaces in SDK V2.2.2 are received through port 10004 while interfaces in SDK V2.1.11 and its previous versions are received through port 10001.

For detailed information, please refer to SDK V2.2.2 user manual.

1.3 Version Features Summary

- In terms of compatibility of this version, the interfaces are basically compatible with the old versions. Some interfaces with compatibility issues will also be explained in the user manual.
- The Python SDK library supports Python 3.10 and above.

For detailed information on SDK V2.2.2, please refer to the corresponding SDK user manuals.

2. Version Features

2.1 New features

2.1.1 Supported Python Versions

The Python SDK library before 2.2.2 only supports Python 3.10 and below. This release has been optimized to support Python versions greater than 3.10. The 3.12.7 used in the test does not have the previous error problem (SystemError: type jkrc.RC has the Py_TPFLAGS_HAVE_GC flag but has no traverse function).

2.2 Function Optimization

2.2.1 Documentation

The TCP and SDK user manuals were checked and improved, and the TCP usage documents and SDK usage documents were transferred online. Users can access them through the web page, URL [JAKA Document Center](https://www.jaka.com/docs/). Among them, the C++ language documents are maintained, and they are explained in detail as much as possible. The documents of other C-like languages, including C and C# languages, only list the interface list. Users can refer to the description of the C++ user manual, the content is basically the same.

2.2.2 Bug fixes

Fixed the problems left over from the previous 1.7.1_40 version of the controller, including the following:

1) After pausing the program, calling the resume program interface reports an error and cannot be restored.
2) Fixed the error after calling IO related interfaces:
    a. get_analog_input,
    b. set_analog_output,
    c. The number of semaphores obtained is abnormal.
3) For arc motion related interfaces, when the number of circles is passed as float type, the interface reports an error.

2.3 Deprecated Features

For design reasons, this release has abandoned some interfaces, including the following categories:

1) Interfaces that have been abandoned because SDK v2.2.2 canceled the polling of port 10004.
    a. set_error_handle (set error callback)
    b. set_status_data_update_time_interval (set the time for SDK to poll port 10004)
    c. set_network_exception_handle (controller action when 10004 is offline)
2) Several interfaces related to trajectory reproduction. These interfaces are cancelled because they have the same functions as FTP-related interfaces. Users can use FTP-related interfaces instead; for the generate_traj_exe_file interface, SDK v2.2.2 will automatically generate a trajectory script when the trajectory recording ends, and

users no longer need to explicitly call this interface.

a. set_block_wait_timeout (set the timeout of the motion instruction block). Note: If this interface function is deleted, the motion will no longer have a timeout limit. If the user wants to stop the motion after sending a very slow motion, he can call the motion_abort interface to stop the motion.
b. get_exist_traj_file_name (get the traj file list)
c. rename_traj_file_name (rename the traj file)
d. remove_traj_exe_file (delete the traj file)
e. generate_traj_exe_file (generate the jks script corresponding to traj)

3) Interfaces that are cancelled due to design requirements and no longer need to be called by users.
   a. set_debug_mode (set the level of detail of print information)
   b. static_Get_SDK_filepath (statically get the SDK log path)
   c. static_Set_SDK_filepath (statically set the SDK log path)
   d. set_debug_mode (set the level of detail of print information)
4) Interfaces that have been abandoned due to error reporting mechanism optimization. SDK v2.2.2 will directly use the error code and error information reported by the controller, so no additional files are required (the above is only for controller errors, not SDK interface errors, and customers need to query based on the jkerr.h file).
   a. set_errorcode_file_path

2.4 Additional Materials

This version adds JAKA SDK Quick Start Instructions (https://www.jaka.com/docs/) to help users build an application using JAKA SDK from 0 to 1. It includes the following parts:

1) Development examples and documentation for building C# application interface programs using Microsoft Visual Studio on Windows;
2) Demo and documentation for creating C++ applications using CMake on Linux
3) Demo and documentation for deploying JAKA SDK demo using QT.
4) Steps and instructions for deploying JAKA SDK applications using the Addon framework, including example steps for building JAKA SDK development programs using CMake.

The corresponding program examples are already included in the Demo folder of the released SDK package. Readers can refer to the relevant content if necessary.