



Kinova.API.Jaco

R5.0.2

Release notes

Version 1.0.0

June 5 2013

Warning

The information contained in this document is the property of Kinova. Except as specifically authorized in writing by Kinova, the holder of this document shall keep the information contained here confidential and shall protect same in whole or in part from disclosure and dissemination to third parties and use same for evaluation, operation and maintenance purposes only. This document is not an engagement of Kinova to develop, implement or design the product or techniques described here.

Revisions & Approval

Version	Primary Author(s)	Description of Version	Date Completed
1.0.0	H. Lamontagne	Initial draft	2013-06-05

Approval History

Approving Party	Version Approved	Signature	Date

Contents

REVISIONS & APPROVAL	2
CONTENTS	3
1 RELEASE NOTES.....	5
1.1 VERSION 4.0.0.....	5
1.1.1 Remove the method <i>ExecuteAPIUnitTests</i>	5
1.1.2 Exception handling improvement.....	5
1.1.3 Add a new method – <i>RealSpeedTest</i>	5
1.2 VERSION 4.0.1.....	5
1.2.1 Security modification	5
1.2.2 Modification of <i>CJacoArm</i> constructor	5
1.2.3 Modification of <i>ValidatePassword</i> method	5
1.2.4 Remove the method <i>ReleaseUSBConnection</i>	5
1.2.5 Remove the class <i>CLogsManagement</i>	5
1.2.6 Remove the mapping serialization during profile backup	6
1.2.7 Serialization/Deserialization method's modification	6
1.2.8 New functionality system.....	6
1.2.9 Activity logging	6
1.2.10 New encryption system	6
1.3 VERSION 4.0.2 (COMPATIBLE ONLY WITH DSP 4.1.6)	6
1.3.1 Correct syntax error – <i>GetPeripheralInformationFromJaco</i>	6
1.3.2 New Ubuntu version	6
1.3.3 <i>CPosition</i> class modification	6
1.3.4 <i>CClientConfigurations</i> class modification.....	6
1.4 VERSION 4.0.3 (COMPATIBLE ONLY WITH DSP 4.1.8 AND FURTHER)	6
1.4.1 New functionality – <i>Send trajectory to Jaco</i>	6
1.5 VERSION 4.0.4 (COMPATIBLE ONLY WITH DSP 4.1.8 AND FURTHER)	7
1.5.1 Functionality modification – <i>Mapping in the profile</i>	7
1.5.2 New functionality – <i>Get Trajectory FIFO information</i>	7
1.5.3 New functionality – <i>Erase the trajectory FIFO</i>	7
1.5.4 Communication improvement	7
1.5.5 Namespace modification.....	7
1.5.6 New constant - <i>CJacoArm</i>	7
1.5.7 New constant - <i>CVectorEuler</i>	7
1.5.8 New constant - <i>CVectorAngle</i>	7
1.6 VERSION 4.0.5 (COMPATIBLE ONLY WITH DSP 4.2.3)	8
1.6.1 Modification – <i>Get info about trajectory FIFO</i>	8
1.6.2 New functionality – <i>GetPositioningInfo</i>	8
1.6.3 New functionality – <i>GetCommandInfo</i>	8
1.6.4 New functionality – <i>GetForceInfo</i>	8
1.6.5 New functionality – <i>GetCurrentInfo</i>	8
1.6.6 New functionality – <i>GetSensorsInfo</i>	8
1.6.7 New functionality – <i>SendBasicTrajectory</i>	8
1.6.8 New functionality – <i>GetSingularityVector</i>	8
1.6.9 New functionality – <i>GetGripperInfo</i>	8

1.7	VERSION 5.0.0 BETA (COMPATIBLE ONLY WITH DSP 4.2.3).....	8
1.7.1	New fonctionnality – Remote API.....	8
1.8	VERSION 5.0.1 BETA (COMPATIBLE ONLY WITH DSP 4.2.5).....	9
1.8.1	New fonctionnality – GetPositionCurrentActuators	9
1.8.2	New fonctionnality – SetActuatorPID	9
1.8.3	New fonctionnality – New C++ version	9
1.9	VERSION 5.0.2 (COMPATIBLE WITH DSP 4.2.5 TO 4.2.9)	10
1.9.1	Virtual joystick bug corrected	10
1.9.2	Installation bug corrected	10

1 Release notes

1.1 Version 4.0.0

1.1.1 Remove the method `ExecuteAPIUnitTests`

This method has been removed because all the tests will now be performed from another project.

1.1.2 Exception handling improvement

- Method `SetClientConfigurations` (`CClientConfigurations`) now manage `ArgumentException` when the `CClientConfigurations` parameter is not valid.
- Method `SetClientConfigurations` (`CClientConfigurations`) now manage `Exception` when a general `Exception` is caught during the process.

1.1.3 Add a new method – `RealSpeedTest`

This method is in the class `CJacoDiagnosticDataManager` and in the class `CCommManager`. It returns a double that represents the quantity of ms to perform 100 transactions of 128 bytes. 12800 bytes total.

1.2 Version 4.0.1

1.2.1 Security modification

The security is now managed by a DLL called `Kinova.DLL.SafeGate`. Check the new code example to know the new security protocol when declaring a `CJacoArm` object.

1.2.2 Modification of `CJacoArm` constructor

The `CJacoArm`'s constructor now takes a `CCypherMessage` object instead of a string.

1.2.3 Modification of `ValidatePassword` method

The method `ValidatePassword` from the class `CPasswordManager` now takes a `CCypherMessage` instead of a string.

1.2.4 Remove the method `ReleaseUSBConnection`

The API had 2 methods in the process of closing the USB connection. Now everything is confined in the single method `CloseConnection` of the Class `CJacoArm` which already exist.

1.2.5 Remove the class `CLogsManagement`

The class `CLogsManagement` has been removed and all of its functionalities are in the class `CJacoDiagnosticDataManager`.

1.2.6 Remove the mapping serialization during profile backup

The mapping serialization has been removed from the profile functionalities.

1.2.7 Serialization/Deserialization method's modification

All serialization and deserialization method of the class CjacoConfigurationManager got their name modified. The serialization's type is no longer specified in the method's name. All the serialization are now binary instead of XML/binary.

1.2.8 New functionality system

All the functionality system has been improve. Does not affect the user but improve API maintenance.

1.2.9 Activity logging

All API activities are now logged in the directory\Application Data\Kinova\Products\Jacosoft\API\Data\Logs\Activities

1.2.10 New encryption system

1.3 Version 4.0.2 (compatible only with DSP 4.1.6)

1.3.1 Correct syntax error – GetPeripheralInformationFromJaco

The method's name has been corrected.

1.3.2 New Ubuntu version

From now on, a Ubuntu version of this project will be available.

1.3.3 CPosition class modification

The CPosition class now contains information about the temperature sensors, the CAN errors of each joint and the controlling time.

1.3.4 CClientConfigurations class modification

The CClientConfigurations now accept a list of 20 CUserPosition to customize the advance retracted position.

1.4 Version 4.0.3 (compatible only with DSP 4.1.8 and further)

1.4.1 New functionality – Send trajectory to Jaco

You can now send a trajectory to Jaco and it will execute it. The method SendTrajectoryFunctionnality(CPointsTrajectory PointsTrajectory) will send a serie of point the represents a trajectory and the method GetActualTrajectoryInfo() will show the information about the current trajectory.

1.5 Version 4.0.4 (compatible only with DSP 4.1.8 and further)

1.5.1 Functionality modification – Mapping in the profile

From now on, the functionalities CreateProfile and LoadProfile from the configuration manager take into consideration the control mapping data structure.

1.5.2 New functionality – Get Trajectory FIFO information

This functionality gives you information about the trajectory FIFO which is inside Jaco. That information include: Quantity of point that are still in the FIFO, the usage percentage and the max size of the FIFO. The functionality can be accessed in the control manager.

1.5.3 New functionality – Erase the trajectory FIFO

This functionality erases all trajectories inside the trajectory FIFO. The functionality can be accessed in the control manager.

1.5.4 Communication improvement

All the communication process has been optimized.

1.5.5 Namespace modification

Some functionality has been moved into their right namespace.

1.5.6 New constant - CJacoArm

New constants (VERSION_DSP, VERSION_JOINT1, VERSION_JOINT2, VERSION_JOINT3, VERSION_JOINT4, VERSION_JOINT5, VERSION_JOINT6, VERSION_FINGER1, VERSION_FINGER2, VERSION_FINGER3, VERSION_CAN1, VERSION_CAN2) were added in the class CJacoArm. Those constants can be used as index in the code version's array returned by the functionality GetCodeVersion.

1.5.7 New constant - CVectorEuler

New constants (COORDINATE_X, COORDINATE_Y, COORDINATE_Z, THETA_X, THETA_Y, THETA_Z) were added in the class CVectorEuler. Those constants can be used as index in the Position array and in the Rotation array.

1.5.8 New constant - CVectorAngle

New constants (JOINT_1, JOINT_2, JOINT_3, JOINT_4, JOINT_5, JOINT_6) were added in the class CVectorEuler. Those constants can be used as index in the Angle array.

1.6 Version 4.0.5 (compatible only with DSP 4.2.3)

1.6.1 Modification – Get info about trajectory FIFO

Now, when you ask the information about the trajectory FIFO, you can know the maximum size of the FIFO.

1.6.2 New functionality – GetPositioningInfo

Get information related to positioning. That includes cartesian and angular position.

1.6.3 New functionality – GetCommandInfo

Get information related to the command sent to Jaco.

1.6.4 New functionality – GetForceInfo

Get all information related to the force.

1.6.5 New functionality – GetCurrentInfo

Get all information related to electric current.

1.6.6 New functionality – GetSensorsInfo

Get all information related to the captors. That includes the supply voltage, the current, the average power, the X – Y – X accelerators and the temperatures.

1.6.7 New functionality – SendBasicTrajectory

Send a trajectory without the limitations.

1.6.8 New functionality – GetSingularityVector

Get a vector that shows the nearest singularity.

1.6.9 New functionality – GetGripperInfo.

Get information about Jaco's gripper.

1.7 Version 5.0.0 beta (compatible only with DSP 4.2.3)

1.7.1 New fonctionnality – Remote API

From now on, you can send command to many Jaco at the same time. All the command pass through a communication server that must be started before using the API.

1.8 Version 5.0.1 beta (compatible only with DSP 4.2.5)

1.8.1 New fonctionnalité – GetPositionCurrentActuators

This functionality lets you get the position and current of each actuator of Jaco. It has been developed to avoid sending 2 commands and save some time.

1.8.2 New fonctionnalité – SetActuatorPID

This functionality let you modify the PID (proportional – integral - derivative) parameters of each actuator by specifying the address of the actuator and the P, I and D value.

1.8.3 New fonctionnalité – New C++ version

A C++ version of the API is now available on Windows and Ubuntu. It includes the following functionalities:

- Get the code version of the main firmware.
- Get the cartesian position.
- Get the angular position.
- Get the cartesian force.
- Get the angular force.
- Get the angular current.
- Get the actual trajectory.
- Get the sensors info.
- Get the global trajectory info.
- Get the singularity vector.
- Get the client configuration.
- Get the position and the current of each actuator.
- Set the robot to angular control.
- Set the robot to cartesian control.
- Get the control of the robot.
- Stop the control of the robot.
- Restore the robot to factory default.
- Send a virtual joystick command.
- Set the client configuration
- Erase all trajectories.
- Set the PID parameter of an actuator.
- Send an advance trajectory.
- Send a basic trajectory.

1.9 Version 5.0.2 (compatible with DSP 4.2.5 to 4.2.9)

This version does not include new functionalities. The code has been adjusted to be compatible with the latest DSP's firmware version and known bugs have been corrected.

1.9.1 Virtual joystick bug corrected

There was a glitch in the virtual joystick's serialization. It has been corrected.

1.9.2 Installation bug corrected

There was a bug regarding the SetPID functionality during the installation of the API. It has been corrected.