

PEAK Robotics Kinedx with Rail



Manufacturer: PEAK Robotics
 Model Number: Peak-Robotics-Kinedx-Rail
 Web Address: www.peakrobotics.com

The Kinedx robots attached to a rail have five-axis and offer a high payload and excellent repeatability, all in a very compact and industrialized package. The innovative Tri-Link arm provides for a super extended reach as well as the ability to fully retract in on itself.

Robotic Arm: Commands

▶ **GetErrorCode(errorCode)** - Retrieves the error message for the error code.

errorCode	Integer	The number of an error. Range of values: 1 to 325.
-----------	---------	--

▶ **GetVersionNumber()** - Reports the release date and the version number for the DLL.

▶ **MotorEnable(axis, enableState)** - Enables or disables the specified axis. When disabled, a motor is free and can be moved by hand.

axis	Integer	Motor to be enabled or disabled. Range of values: 1 to 5.
enableState	Boolean	A value of True will enable the motor, and a value of False will disable the motor.

▶ **MoveAbsoluteAllAxes(axis1, axis2, axis3, axis4, axis5, cmdVel, cmdAccel)** - Moves all axes at once to a specific absolute position.

axis1	Float	Value of axis 1.
axis2	Float	Value of axis 2.
axis3	Float	Value of axis 3.
axis4	Float	Value of axis 4.
axis5	Float	Value of axis 5.
cmdVel	Float	Velocity of the move expressed in percentage of maximum. Range of values: 1.0 to 100.0.
cmdAccel	Float	Acceleration of the move, expressed in percentage of maximum. Range of values: 1.0 to 100.0.

▶ **MoveAbsoluteSingleAxis(axis, position, vel, accel)** - Moves a motor to a specific absolute position.

axis	Integer	Motor to be moved. Range of values: 1 to 5.
position	Float	Absolute destination position of the move, expressed in degrees or millimeters.
vel	Float	Velocity of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.
accel	Float	Acceleration of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.

▶ **MoveRelativeCartesian(xMoveDist, yMoveDist, zMoveDist, thetaMoveDist, cmdVel, cmdAccel)** - Moves the robot a relative distance in a Cartesian coordinate from the current position of the robot.

xMoveDist	Float	X Cartesian coordinate offset from the current position of the robot.
yMoveDist	Float	Y Cartesian coordinate offset from the current position of the robot.
zMoveDist	Float	Z Cartesian coordinate offset from the current position of the robot.
thetaMoveDist	Float	Theta Cartesian coordinate offset from the current position of the robot.
cmdVel	Float	Velocity of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.
cmdAccel	Float	Acceleration of the move, expressed in percent of maximum. Range of values: 1.0 to 100.0.

▶ **MoveRelativeSingleAxis(axis, distance, vel, accel)** - Moves a motor a specific relative distance from its current position.

axis	Integer	Motor to be moved. Range of values: 1 to 5.
distance	Float	Relative distance for the move, expressed in degrees or millimeters.
vel	Float	Velocity of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.
accel	Float	Acceleration of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.

▶ **MoveToCartesian(xCoordinate, yCoordinate, zCoordinate, thetaCoordinate, cmdVel, cmdAccel)** - Moves the robot to an absolute Cartesian coordinate.

xCoordinate	Float	X coordinate value to which the robot will be moved.
yCoordinate	Float	Y coordinate value to which the robot will be moved.
zCoordinate	Float	Z coordinate value to which the robot will be moved.

thetaCoordinate	Float	Theta coordinate value to which the robot will be moved.
cmdVel	Float	Velocity of the move expressed in percentage of maximum. Range of values: 1.0 to 100.0.
cmdAccel	Float	Acceleration of the move, expressed in percentage of maximum. Range of values: 1.0 to 100.0.

► **PlacePlateInHotel(topTeachPoint, bottomTeachPoint, retractTeachPoint, hotelCapacity, plateNumber, liftHeightMM, velocity, gripperTimeDelayMsec)** - Places a plate in the desired slot of a random-access hotel. The hotel is defined by three teach points and by the number of plates that can be loaded into the hotel. The robot will move to above and retracted from the hotel slot, into the slot, down to the slot, and release the plate from the gripper. The robot will then move to above the plate, and then retracts out of the slot.

topTeachPoint	String	Teach point that has been taught at the top slot of the hotel.
bottomTeachPoint	String	Teach point that has been taught at the bottom slot of the hotel.
retractTeachPoint	String	Teach point that has been taught at the retract position. The retract position should be taught at the same Z height as the TopTeachPoint, with the robot retracted out of the hotel.
hotelCapacity	Integer	Total number of slots in the hotel.
plateNumber	Integer	Slot in the hotel where the plate currently in the gripper will be placed.
liftHeightMM	Float	Height in millimeters the robot must move above the current hotel slot when moving into and out of the slot.
velocity	Float	Velocity of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.
gripperTimeDelayMsec	Integer	Number of milliseconds to wait after opening the gripper (give the gripper enough time to open before the robot moves away).

► **PlacePlateInStack(topTeachPoint, bottomTeachPoint, stackCapacity, plateNumber, topClearanceMM, velocity, gripperTimeDelayMsec)** - Places a plate at a desired height on a stack of plates. The plate stack is defined by two teach points and by the number of plates that are in the stack. The robot will move to above the stack, move down to the specified height, and release the plate from the gripper. The robot will then move to above the stack.

topTeachPoint	String	Teach point that has been taught at the top slot of the hotel.
bottomTeachPoint	String	Teach point that has been taught at the bottom slot of the hotel.
stackCapacity	Integer	Maximum number of plates that can be placed in the stack.
plateNumber	Integer	Position in the stack where the plate currently in the gripper will be placed.

topClearanceMM	Float	Height in millimeters that the robot must move above the top of the stack when approaching and departing from the stack.
velocity	Float	Velocity of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.
gripperTimeDelayMsec	Integer	Number of milliseconds to wait after opening the gripper (give the gripper enough time to open before the robot moves away).

▶ **GetJoint()** - Returns the current position of the arm in joint coordinates.

▶ **GetPoint()** - Returns the current position of the arm in cartesian coordinates.

▶ **ReadInput(axis, inputNumber)** - Reads the state of any input. Each motor has four digital inputs. The first two inputs on each motor are used for serial communication except for the wrist motor (axis 4), which only uses input 1 for serial communication. Each home sensor is connected to input 3 on the corresponding motor. There are also four auxiliary inputs that can be accessed using the auxiliary connector at the base of the robot.

axis	Integer	Specifies the motor. Use Axis = 0 for auxiliary inputs. Range of values: 0 to 5.
inputNumber	Integer	Input number on the specified motor. The gripper sensor is connected to axis 2 and input 4 (output 2 must be turned on before the gripper sensor can be read).

▶ **RemovePlateFromHotel(topTeachPoint, bottomTeachPoint, retractTeachPoint, hotelCapacity, plateNumber, liftHeightMM, velocity, gripperTimeDelayMsec)** - Removes a plate from the desired slot of a random access hotel. The hotel is defined by three teach points and by the number of plates that fit in the hotel. The robot will move to above and retracted from the slot, into the slot, down to the slot, and will grip the plate in the slot. The robot will then lift the plate, and then retract out of the slot.

topTeachPoint	String	Teach point that has been taught at the top slot of the hotel.
bottomTeachPoint	String	Teach point that has been taught at the bottom slot of the hotel.
retractTeachPoint	String	Teach point that has been taught at the retract position. The retract position should be taught at the same Z height as the TopTeachPoint, with the robot retracted out of the hotel.
hotelCapacity	Integer	Total number of slots in the hotel.
plateNumber	Integer	Slot in the hotel where the plate currently in the gripper will be placed.
liftHeightMM	Float	Height in millimeters the robot must move above the current hotel slot when moving into and out of the slot.
velocity	Float	Velocity of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.

gripperTimeDelayMsec	Integer	Number of milliseconds to wait after opening the gripper (give the gripper enough time to open before the robot moves away).
----------------------	---------	--

► **RemovePlateFromStack(topTeachPoint, bottomTeachPoint, stackCapacity, plateNumber, topClearanceMM, velocity, searchForPlate, sensorOffsetMM, gripperTimeDelayMsec)** - Remove a plate at a desired height from a stack of plates, or the gripper sensor can be used to detect the position of the top plate. The plate stack is defined by two teach points and by the number of plates that are in a full stack. The robot will move to above the stack, move down to the plate, grip it, and lift it out of the stack.

topTeachPoint	String	Teach point that has been taught at the top slot of the hotel.
bottomTeachPoint	String	Teach point that has been taught at the bottom slot of the hotel.
stackCapacity	Integer	Maximum number of plates that can be placed in the stack.
plateNumber	Integer	Position in the stack where the plate to be removed is currently located.
topClearanceMM	Float	Height in millimeters that the robot must move above the top of the stack when approaching and departing from the stack.
velocity	Float	Velocity of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.
searchForPlate	Boolean	Set to True to use the sensor to locate the height of the plate. Set to False to use the height calculated from the PlateNumber.
sensorOffsetMM	Float	Once the gripper sensor detects the plate and the robot stops, the robot will move the amount specified by this parameter. The robot may not stop at exactly the right grip height, so this parameter can be used to adjust the height of the robot before gripping the plate.
gripperTimeDelayMsec	Integer	Number of milliseconds to wait after opening the gripper (give the gripper enough time to open before the robot moves away).

► **ScanHotel(topTeachPoint, bottomTeachPoint, retractTeachPoint, hotelCapacity, plateNumber, liftHeightMM, velocity)** - Determines whether a plate is present at a specified slot.

topTeachPoint	String	Teach point that has been taught at the top slot of the hotel.
bottomTeachPoint	String	Teach point that has been taught at the bottom slot of the hotel.
retractTeachPoint	String	Teach point that has been taught at the retract position. The retract position should be taught at the same Z height as the TopTeachPoint, with the robot retracted out of the hotel.

hotelCapacity	Integer	Total number of slots in the hotel.
plateNumber	Integer	Slot in the hotel to be scanned.
liftHeightMM	Float	Height in millimeters the robot must move above the current hotel slot when moving into and out of the slot.
velocity	Float	Velocity of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.

▶ **ScanStack(topTeachPoint, bottomTeachPoint, stackCapacity, topClearanceMM, velocity, sensorOffsetMM)** - Determine the number of plates currently in the stack.

topTeachPoint	String	Teach point that has been taught at the top slot of the hotel.
bottomTeachPoint	String	Teach point that has been taught at the bottom slot of the hotel.
stackCapacity	Integer	Maximum number of plates that can be placed in the stack.
topClearanceMM	Float	Height in millimeters that the robot must move above the top of the stack when approaching and departing from the stack.
velocity	Float	Velocity of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.
sensorOffsetMM	Float	Once the gripper sensor detects the plate and the robot stops, the robot will move the amount specified by this parameter. The robot may not stop at exactly the right grip height, so this parameter can be used to adjust the height of the robot before measuring the stack height.

▶ **ScriptRun(filename, scriptName)** - Run a script contained in a file. A script is a series of robot operations saved to a text file.

filename	String	Location of the file in which the script is located. If a blank string is passed in, the default parameter file will be used.
scriptName	String	Name of the script to be run.

▶ **SetOutput(outputNumber, state)** - Controls digital outputs.

outputNumber	Integer	Output to be controlled. 1: gripper valve (false=gripped); 2: gripper sensor power (false=disabled); 3: auxiliary output. Range of values: 1 to 3.
state	Boolean	Specifies the state (true or false) of the digital output.

▶ **Shutdown()** - Turns off the robot motors and terminates the COM port connection.

▶ **TeachPointMoveRelativeTo(positionNameOrNumber, shoulderOffset, zOffset, elbowOffset, wristOffset, railOffset, velocity, acceleration)** - Moves the robot to the specified Teach Point, with each axis having an offset.

positionNameOrNumber	String	Name or number of the Teach Point to which the robot will move.
----------------------	--------	---

shoulderOffset	Float	Shoulder axis offset.
zOffset	Float	Z axis offset.
elbowOffset	Integer	Elbow axis offset.
wristOffset	Float	Wrist axis offset.
railOffset	Float	Rail axis offset.
velocity	Float	Velocity of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.
acceleration	Float	Acceleration of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.

▶ **TeachPointMoveTo(positionNameOrNumber, velocity, acceleration)** - Moves the robot to the specified Teach Point.

positionNameOrNumber	String	Name or number of the Teach Point to which the robot will move.
velocity	Float	Velocity of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.
acceleration	Float	Acceleration of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.

▶ **TeachPointMoveToLinearIncremental(positionNameOrNumber, velocity, acceleration, numIncrements)** - Moves the robot to the specified Teach Point. The move will be broken up into several smaller moves that cause the end of the robot arm to follow a linear path. The higher the value of NumIncrements, the closer the move will be to following the linear path, but the move will take more time.

positionNameOrNumber	String	Name or number of the Teach Point to which the robot will move.
velocity	Float	Velocity of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.
acceleration	Float	Acceleration of the move expressed in percent of maximum. Range of values: 1.0 to 100.0.
numIncrements	Integer	Number of incremental moves the robot will make on its way to the Teach Point.

▶ **TeachPointsLoad(fileName)** - Reads a file containing Teach Point data. This must be done before the robot can be moved to Teach Points.

fileName	String	Path and name of the file containing the Teach Point data. If left blank, the file path specified in TeachPendant/Tools/Options/File Locations will be used.
----------	--------	--

Robotic Arm: Errors

 **ArmError(code, description)** - Error occurred during command execution.

code	Integer	Error code.
description	String	Error description.

ReTiSoft Inc.
366 Revus Avenue, Unit 21
Mississauga, Ontario, Canada, L5G-4S5
Main: 647-724-2398 Europe: 33-9-7518-0225
Web: www.retisoft.ca Email: prodziew@retisoft.ca