ChoreoLib

LabVIEW

Reference

Version 2024.1.0

Table of Contents

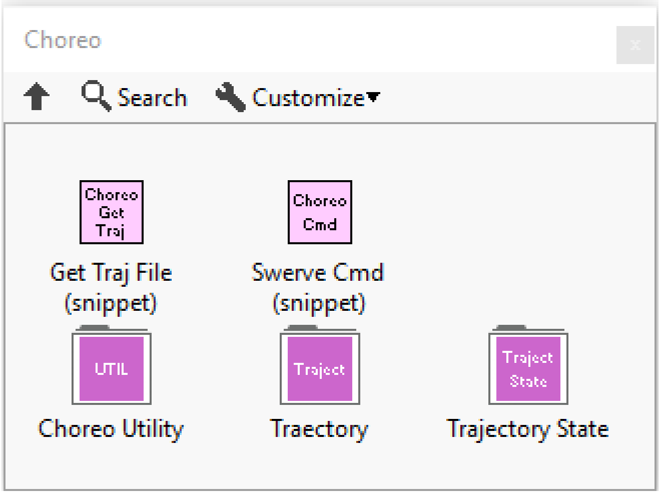
Introduction

The ChoreoLib LabVIEW library provides utility functions to read, sample, and follow Choreo trajectories.

The library source code, package build specifications, and test package can be found here https://github.com/jsimpso81/ChoreoLabVIEW

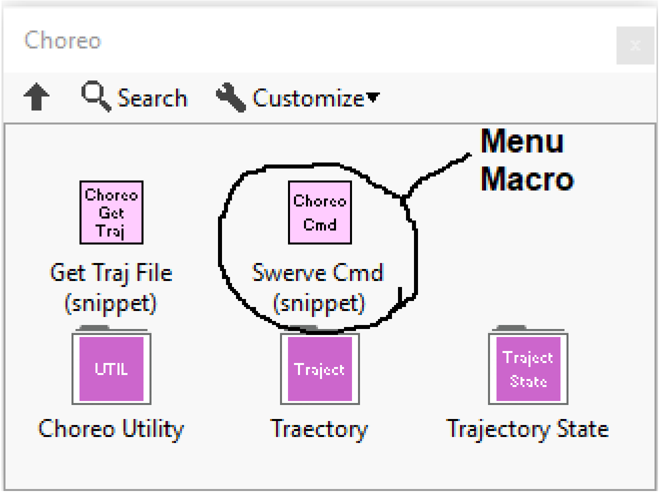
Function Menus

A Choreo function palette contains the Choreo functions and type definitions. This palette can be accessed from the WPI Robotics Library Third Party pallete.



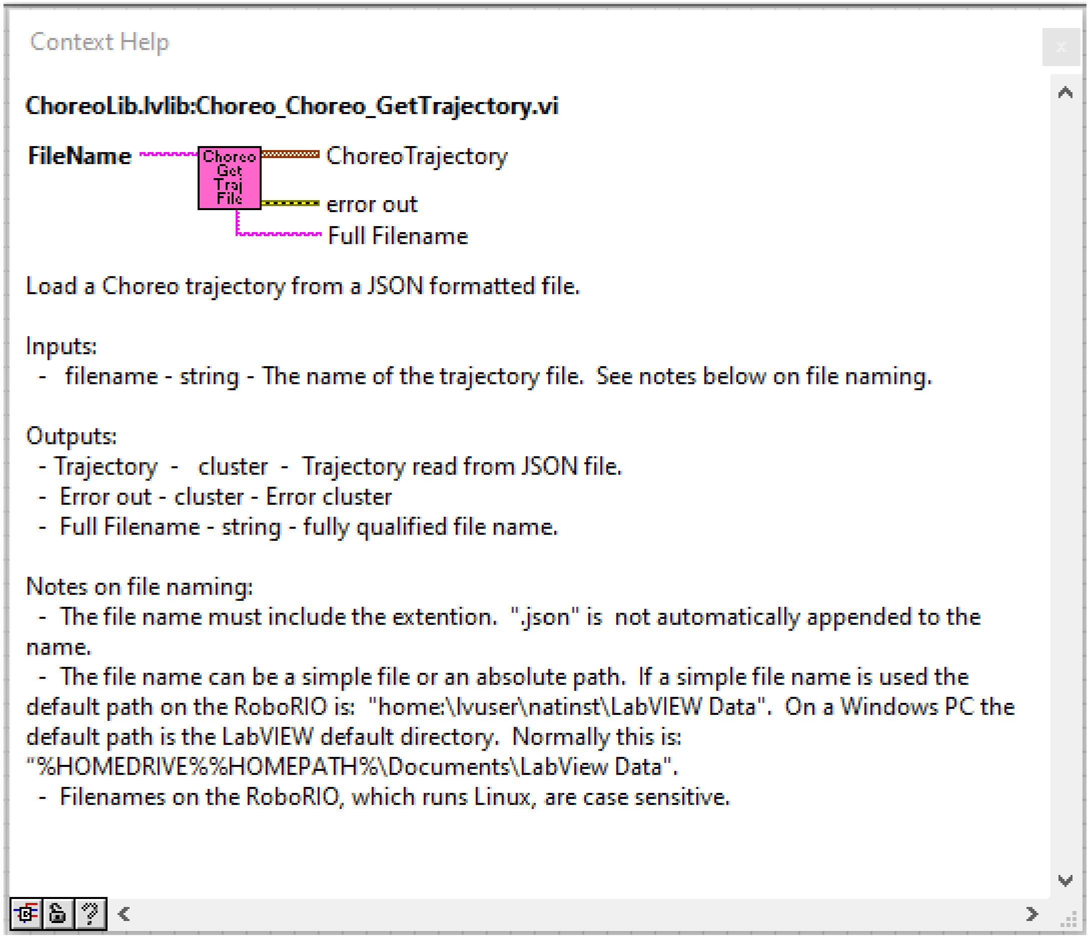
Menu Macros

Some of the menu items place "snippets" of code into a VI. These "macros" can greatly speed up development by placing large sections of mostly completed code in a VI. Usually macros have a different color menu palette icon and may contain "(snippet)" in the description.



Function Help

Each VI includes help that can be accessed using the standard LabVIEW help toggle (Ctrl H).



ChoreoLib.lvlib:Choreo_Choreo_GetTrajectory.vi

Load a Choreo trajectory from a JSON formatted file.

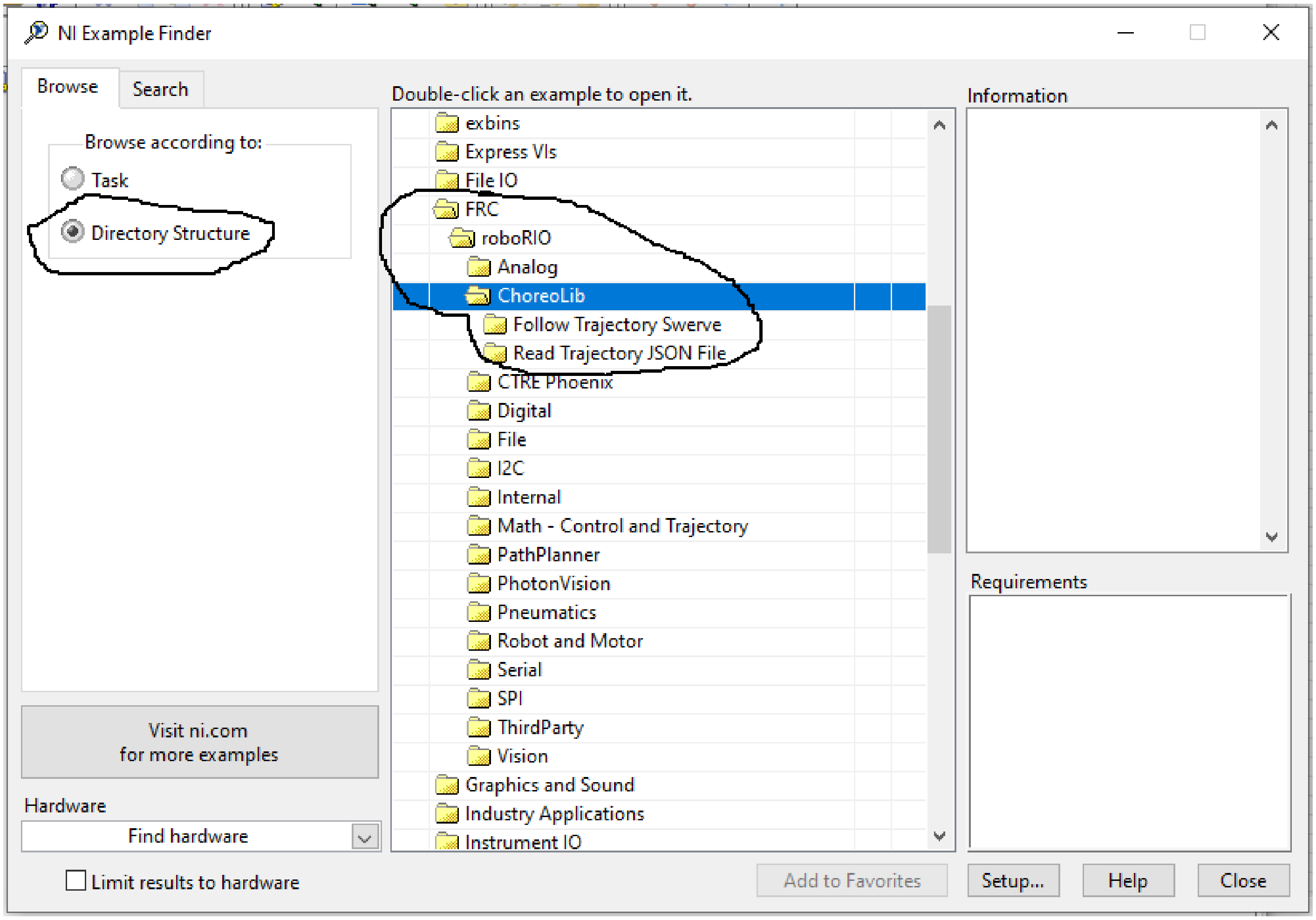
Inputs:
- filename - string - The name of the trajectory file. See notes below on file naming.

Outputs:
- Trajectory -   cluster - Trajectory read from JSON file.
- Error out - cluster - Error cluster
- Full Filename - string - fully qualified file name.

Notes on file naming:
- The file name must include the extention. ".json" is not automatically appended to the name.
- The file name can be a simple file or an absolute path. If a simple file name is used the default path on the RoboRIO is: "home:\lvuser\natinst\LabVIEW Data". On a Windows PC the default path is the LabVIEW default directory. Normally this is: "%HOMEDRIVE%%HOMEPATH%\Documents\LabView Data".
- Filenames on the RoboRIO, which runs Linux, are case sensitive.

Function Examples

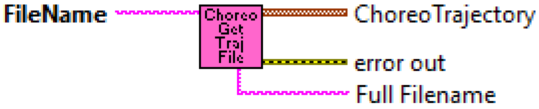
Many of the functions have examples that can be found under the LabVIEW "Find examples..." function. (Help -> Find Examples...). The function examples are easiest to find when "Directory Structure" is selected.


NI Example Finder
Browse
Search
Double-click an example to open it.
Information
Browse according to:
○ Task
● Directory Structure
exbins
Express VIs
File IO
FRC
roboRIO
Analog
ChoreoLib
Follow Trajectory Swerve
Read Trajectory JSON File
CTRE Phoenix
Digital
File
I2C
Internal
Math - Control and Trajectory
PathPlanner
PhotonVision
Pneumatics
Robot and Motor
Serial
SPI
ThirdParty
Vision
Graphics and Sound
Industry Applications
Instrument IO
Requirements
Visit ni.com
for more examples
Hardware
Find hardware
□ Limit results to hardware
Add to Favorites
Setup...
Help
Close

Function Groups

Choreo

Choreo_Choreo_GetTrajectory



Load a Choreo trajectory from a JSON formatted file.   Note that for newer versions of Choreo, the file has a ".traj" extension.   Internally this is formatted as a JSON file.

Inputs:

- filename - string - The name of the trajectory file. See notes below on file naming.

Outputs:

- Trajectory -   cluster - Trajectory read from JSON file.

- Error out - cluster - Error cluster

- Full Filename - string - fully qualified file name.

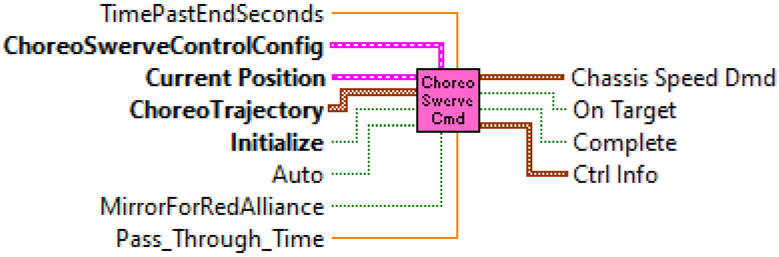
Notes on file naming:

- The file name must include the extention. ".json", or ".traj" is not automatically appended to the name.

- The file name can be a simple file or an absolute path. If a simple file name is used the default path on the RoboRIO is: "home:\lvuser\natinst\LabVIEW Data". On a Windows PC the default path is the LabVIEW default directory. Normally this is: %HOMEDRIVE%%HOMEPATH%\Documents\LabView Data".

- Filenames on the RoboRIO, which runs Linux, are case sensitive.

Choreo_Choreo_SwerveCommand



Command to assist in the execution of a Choreo Trajectory. This wraps the ChoreoSwerveCtrl VI.

Inputs:

- TimePastEndSeconds - double - Number of seconds past the end of the trajectory to continue to control robot position. (Optional. Default: 3.0)

- ChoreoSwerveControlConfig - cluster - Controller configuration data

- Current Pose - pose2d - Current robot position pose calcualted by either Odometry or Pose Estimation.

- ChoreoTrajectory - cluster - Choreotrajectory list

- Initialize - boolean - Set to true for one scan cycle when starting a trajectory. Resets the elapsed time and controllers.

- Auto - boolean - When TRUE closed loop control is used to control trajectory execution. This is mostly for debugging and robot tuning. (Optional. Default: TRUE)

- Mirror For Red Alliance - boolean - Whether or not to mirror the path based on alliance (this assumes the path is created for the blue alliance) If TRUE, the alliance color is queried and if RED, the trajectory is flipped for execution.

- PassThroughTime - double - Continueually counting time (seconds) read from the FGPA on roboRIO or local computer when running on PC. (Optional: Default: Read FGPA time)

Outputs:

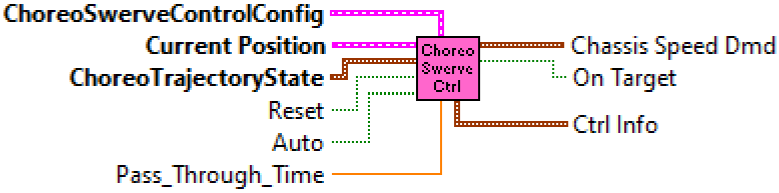
- Choreo Speed Demand - chassis speed - Desired chassis speed (M/S, Rad/Sec)

- On Target - boolean - TRUE when robot position is within tolerance of trajectory position

- Complete - boolean - TRUE when trajectory time has elapsed and robot is on target or when trajectory time is "TimePastEndSeconds" past trajectory end time regardless of On Target status.

- Ctrl Info - cluster - Information on control. Mostly for diagnostics.

Choreo_Choreo_SwerveController



Creates and executes a swerve drive control function to execute a Choreo Trajectory sample.   The controller consists of three advanced PID controllers, one for X position, Y position, and Rotation.

Inputs:

- ChoreoSwerveControlConfig - cluster - Controller configuration data

- Current Pose - pose2d - Current robot position pose calcualted by either Odometry or Pose Estimation.

- ChoreoTrajectoryState - cluster - Trajectory state to execute

- Reset - boolean - Resets the controllers.

- Auto - boolean - When TRUE closed loop control is used to control trajectory execution. This is mostly for debugging and robot tuning. (Optional. Default: TRUE)

- PassThroughTime - double - Continueually counting time (seconds) read from the FGPA on roboRIO or local computer when running on PC. (Optional: Default: Read FGPA time)

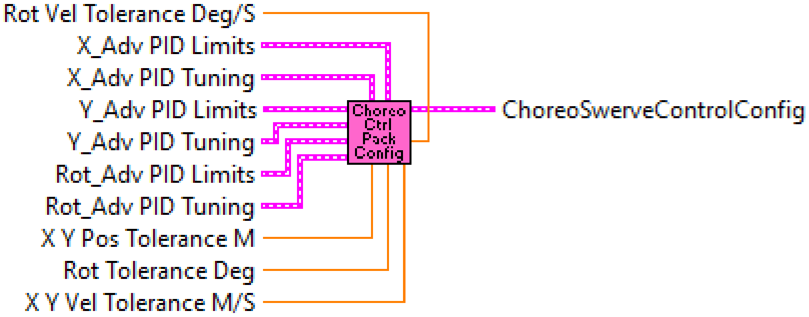
Outputs:

- Choreo Speed Demand - chassis speed - Desired chassis speed (M/S, Rad/Sec)

- On Target - boolean - TRUE when robot position is within tolerance of trajectory position

- Ctrl Info - cluster - Information on control. Mostly for diagnostics.

Choreo_Choreo_SwerveControllerPackConfig



Packs the configuration information for a Choreo Swerve Controller into a cluster for use by the Choreo Swerve Controller or the Choreo Swerve Command functions.

Note:

- Defaults have been chosen for the inputs, however tuning based on actual performance is advised.

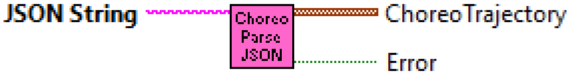
Inputs:

- X Adv PID Limits - cluster - Input and output limits for the X position PID (Optional.)
- X Adv PID Tuning - cluster - Tuning values for the X position PID (optional)
- Y Adv PID Limits - cluster - Input and output limits for the Y position PID (Optional.)
- Y Adv PID Tuning - cluster - Tuning values for the Y position PID (optional)
- Rot Adv PID Limits - cluster - Input and output limits for the rotation position PID. Note this PID must be set to continuous (Optional.)
- Rot Adv PID Tuning - cluster - Tuning values for the rotational position PID (optional)
- X Y Pos Tolerance - double - Distance (Meters) to be considered on target. (Defaultt 0.0381)
- Rot Pos Tolerance - double - Rotation (Degrees) to be considered on target. (Defaultt 10.0)
- X Y Vel Tolerance - double - Velocity (Meters/Second) to be considered on target. (Defaultt 1.0)
- Rot Vel Tolerance - double - Velocity (Degrees/Second) to be considered on target. (Defaultt 30.0)

Outputs:

- ChoreoSwerveControlConfig - cluster - packed controller configuration.

Choreo_Choreo_TrajectoryFromJSON



Parse a choreo trajectory from a JSON formatted string.

Inputs:

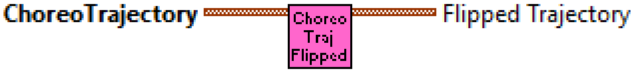
- JSON String - string- The string containing the choreo trajectory. Note that files may have a .TRAJ extension. These are actually internally formatted as JSON files.

Outputs:

- trajectory - cluster - Trajectory read from JSON file.
- Error - boolean - TRUE if an error occured.

Trajectory

Choreo_Trajectory_Flipped



trajectory, mirrored across the field midline.

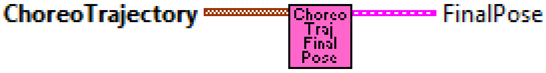
Inputs:

- ChoreoTrajectory - cluster - Input trajectory

Outputs:

- Flipped Trajectory - cluster - trajectory, mirrored across the field midline.

Choreo_Trajectory_GetFinalPose



Return final pose of this trajectory

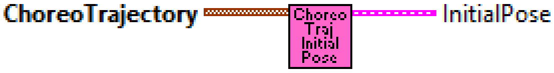
Inputs:

- ChoreoTrajectory - cluster - Input trajectory

Outputs:

- Final Pose - pose2d - the final, non-mirrored pose of the trajectory.

Choreo_Trajectory_GetInitialPose



Return initial pose of this trajectory

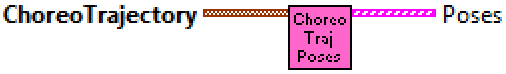
Inputs:

- ChoreoTrajectory - cluster - Input trajectory

Outputs:

- Initial Pose - pose2d - the initial, non-mirrored pose of the trajectory.

Choreo_Trajectory_GetPoses



Return all posse of this trajectory

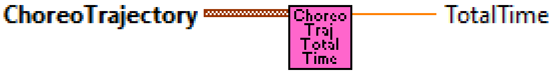
Inputs:

- ChoreoTrajectory - cluster - Input trajectory

Outputs:

- Poses - pose2d array - all, non-mirrored poses of the trajectory.

Choreo_Trajectory_GetTotalTime



Return total time duration of this trajectory

Inputs:

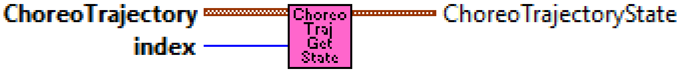
- ChoreoTrajectory - cluster - Input trajectory

Outputs:

- TotalTime - double - The total time duration of this trajectory.

.

Choreo_Trajectory_GetTrajState



Return a single trajectory state from the trajectory.

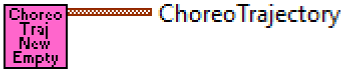
Inputs:

- ChoreoTrajectory - cluster - Input trajectory
- Index - integer - index into the trajectory state array. Beginning index is zero.

Outputs:

- Trajectory state - cluster - the selectred trajectory state..

Choreo_Trajectory_New_Empty



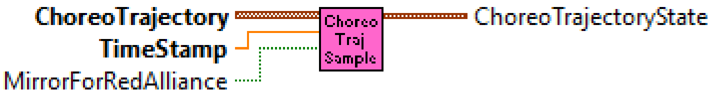
Create an empty trajectory

Inputs:

Outputs:

- ChoreoTrajectory - array - Empty array of Trajectory states.

Choreo_Trajectory_Sample



Return an interpolated sample of the trajectory at the given timestamp.

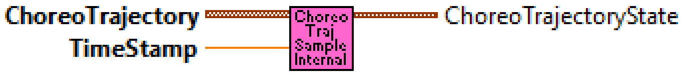
Inputs:

- ChoreoTrajectory - cluster - The trajectory to sample.
- timestamp - double - The timestamp of this sample relative to the beginning of the trajectory.
- mirrorForRedAlliance - boolean - whether or not to return the sample as mirrored across the field midline (as in 2023). (Optional. Default: false)

Outputs:

- ChoreoTrajectoryState - cluster - The ChoreoTrajectoryState at the given time.

Choreo_Trajectory_SampleInternal



This is an internal routine. It should NOT be called by the end user. Return an interpolated sample of the trajectory at the given timestamp.

Inputs:

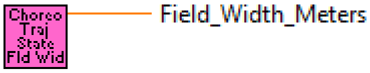
- ChoreoTrajectory - cluster - The trajectory to sample.
- timestamp - double - The timestamp of this sample relative to the beginning of the trajectory.

Outputs:

- ChoreoTrajectoryState - cluster - The ChoreoTrajectoryState at the given time.

TrajectoryState

Choreo_TrajectoryState_FieldWidth



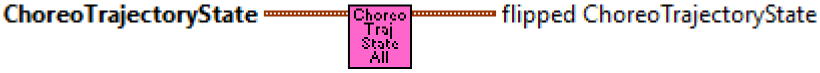
Return FRC field width. This is valid for the current game only.

Inputs:

Outputs:

- Field_Width_Meters - double - Playing field width. meters

Choreo_TrajectoryState_Flipped



Return "flipped:" trajectory state, mirrored accros the field mid-line.

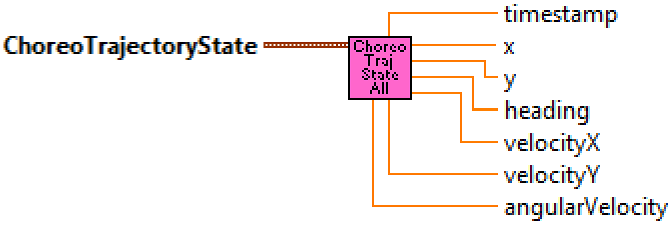
Inputs:

- ChoreoTrajectoryState - cluster - Individual trajectory state

Outputs:

- Flipped_ChoreoTrajectoryState - cluster -this state, mirrored across the field midline.

Choreo_TrajectoryState_GetAll



Return individual data from a ChoreoTrajectoryState cluster

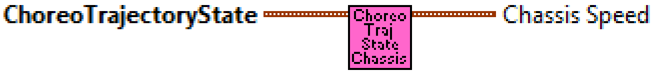
Inputs:

- ChoreoTrajectoryState - cluster - Individual trajectory state

Outputs:

- timestamp - double - The timestamp of this state, relative to the beginning of the trajectory.
- x - double - The X position of the state in meters.
- y - double - The Y position of the state in meters.
- heading - double - The heading of the state in radians, with 0 being in the +X direction.
- velocityX - double - The velocity of the state in the X direction in m/s.
- velocityY - double - The velocity of the state in the X direction in m/s.
- angularVelocity - double - The angular velocity of the state in rad/s.

Choreo_TrajectoryState_GetChassisSpeeds



Return desired chassis speed from the trajectory state. This can be used as a feed forward for control.

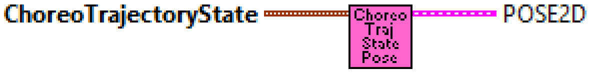
Inputs:

- ChoreoTrajectoryState - cluster - Individual trajectory state

Outputs:

- ChassisSpeeds - cluster - the field-relative chassis speeds of this state.

Choreo_TrajectoryState_GetPose



Return the pose2d of this state.

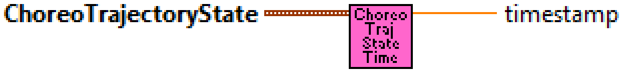
Inputs:

- ChoreoTrajectoryState - cluster - Individual trajectory state

Outputs:

- Pose2d - cluster - the pose at this state.

Choreo_TrajectoryState_GetTime



Return the timestamp of this state. This is the elapsed seconds from the start of the trajectory.

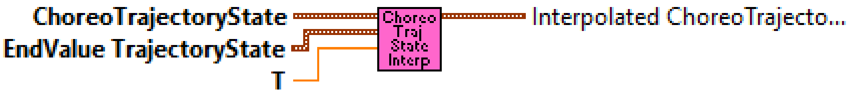
Inputs:

- ChoreoTrajectoryState - cluster - Individual trajectory state

Outputs:

- timestamp - double - The timestamp of this state. Seconds.

Choreo_TrajectoryState_Interpolate



Interpolate between two trajectory states.

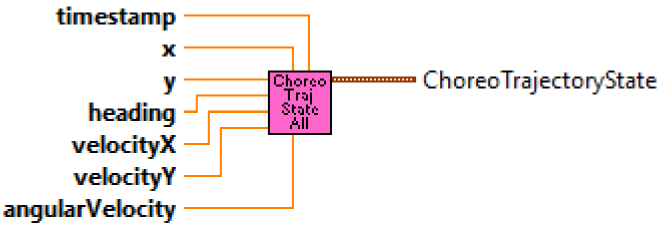
Inputs:

- ChoreoTrajectoryState - cluster - The earlier trajectory state.
- EndValue_TrajectoryState - cluster - The next state. It should have a timestamp after this state.
- T - double - the timestamp of the interpolated state. It should be between this state and endValue.

Outputs:

- InterpolatedState - cluster - Interpolated trajectory state.

Choreo_TrajectoryState_New



Create a ChoreoTrajectoryState cluster from individual elements.

Inputs:

- timestamp - double - The timestamp of this state, relative to the beginning of the trajectory.
- x - double - The X position of the state in meters.
- y - double - The Y position of the state in meters.
- heading - double - The heading of the state in radians, with 0 being in the +X direction.
- velocityX - double - The velocity of the state in the X direction in m/s.
- velocityY - double - The velocity of the state in the X direction in m/s.

- angularVelocity - double - The angular velocity of the state in rad/s.

Outputs:

- ChoreoTrajectoryState - cluster - Individual trajectory state

Type Definitions

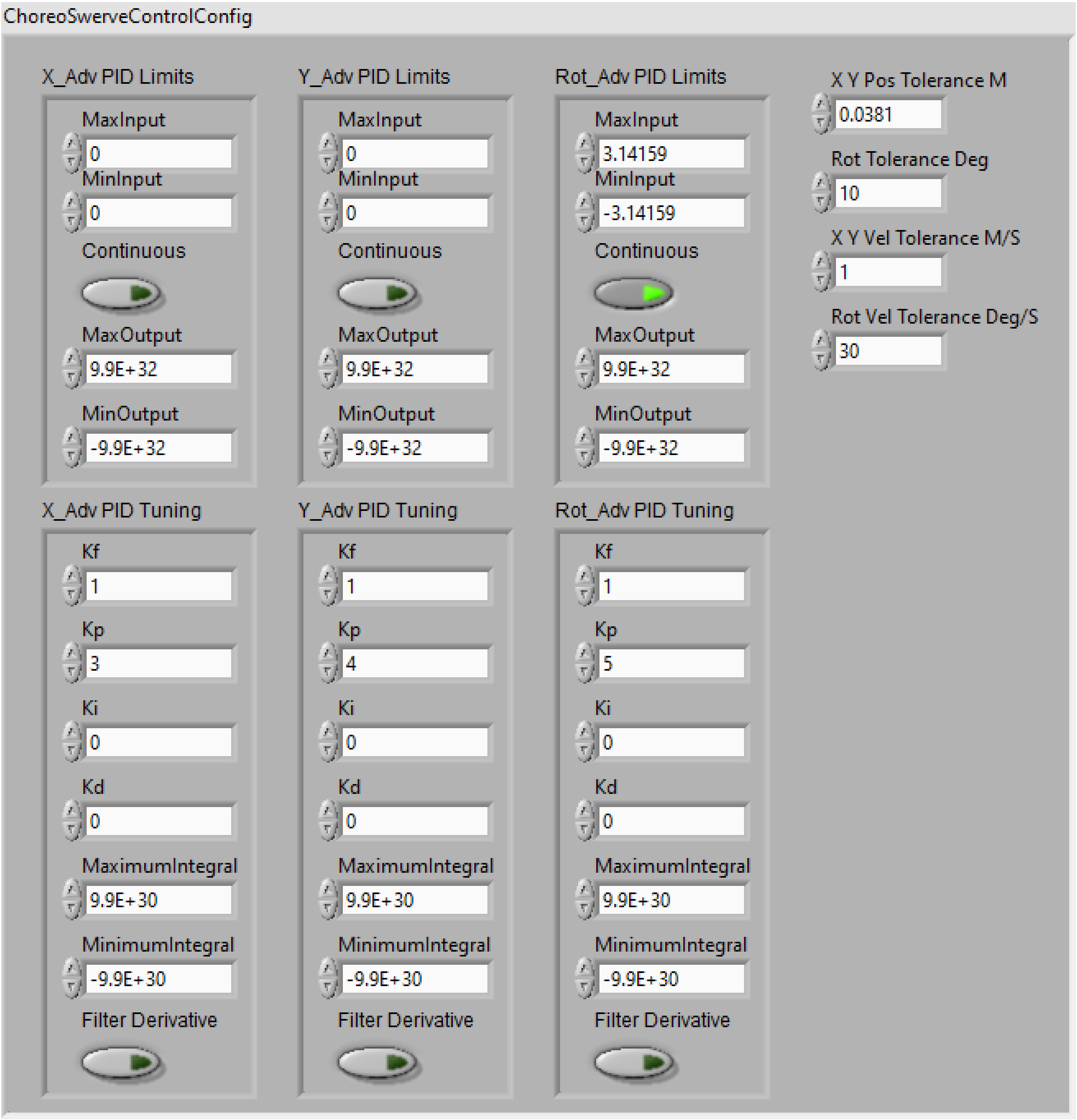
TypeDef

TypeDef-ChoreoSwerveControlConfig

Choreo Swerve Control Config

Controller configuration cluster for the ChoreoSwerveController

This cluster contains the configuration data for the 3 advanced PIDs.   This includes limits, tuning, and tolerance configuration.



ChoreoSwerveControlConfig

X_Adv PID Limits

MaxInput: 0
MinInput: 0
Continuous
MaxOutput: 9.9E+32
MinOutput: -9.9E+32

Y_Adv PID Limits

MaxInput: 0
MinInput: 0
Continuous
MaxOutput: 9.9E+32
MinOutput: -9.9E+32

Rot_Adv PID Limits

MaxInput: 3.14159
MinInput: -3.14159
Continuous
MaxOutput: 9.9E+32
MinOutput: -9.9E+32

X Y Pos Tolerance M: 0.0381

Rot Tolerance Deg: 10

X Y Vel Tolerance M/S: 1

Rot Vel Tolerance Deg/S: 30

X_Adv PID Tuning

Kf: 1
Kp: 3
Ki: 0
Kd: 0
MaximumIntegral: 9.9E+30
MinimumIntegral: -9.9E+30
Filter Derivative

Y_Adv PID Tuning

Kf: 1
Kp: 4
Ki: 0
Kd: 0
MaximumIntegral: 9.9E+30
MinimumIntegral: -9.9E+30
Filter Derivative

Rot_Adv PID Tuning

Kf: 1
Kp: 5
Ki: 0
Kd: 0
MaximumIntegral: 9.9E+30
MinimumIntegral: -9.9E+30
Filter Derivative

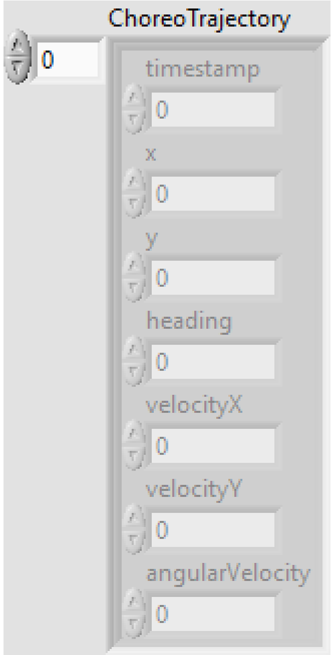
TypeDef-ChoreoTrajectory

Choreo Traj

A trajectory loaded from Choreo.

Contains:

- ChoreoTrajectory - array of ChoreoTrajectoryState



ChoreoTrajectory

0

timestamp 0

x 0

y 0

heading 0

velocityX 0

velocityY 0

angularVelocity 0

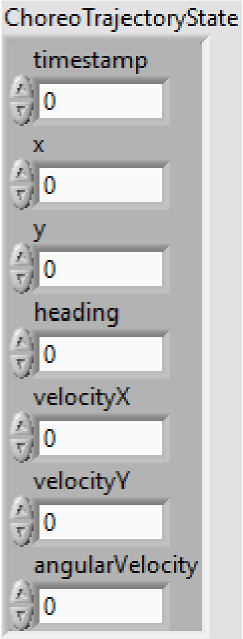
TypeDef-ChoreoTrajectoryState

Choreo
Traj
State

Choreo Trajectory State

Contains:

- timestamp - double - The timestamp of this state, relative to the beginning of the trajectory.
- x - double - The X position of the state in meters.
- y - double - The Y position of the state in meters.
- heading - double - The heading of the state in radians, with 0 being in the +X direction.
- velocityX - double - The velocity of the state in the X direction in m/s.
- velocityY - double - The velocity of the state in the X direction in m/s.
- angularVelocity - double - The angular velocity of the state in rad/s.



ChoreoTrajectoryState

timestamp: 0

x: 0

y: 0

heading: 0

velocityX: 0

velocityY: 0

angularVelocity: 0

Enumerated Type Definitions