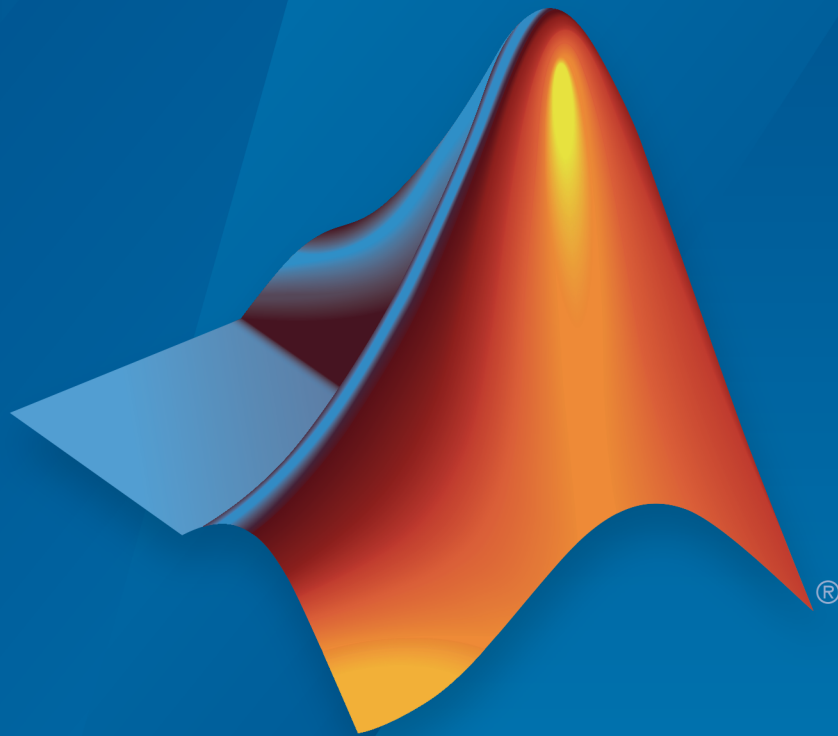


SimMechanics™

Reference



MATLAB® & SIMULINK®

R2015a



## How to Contact MathWorks



Latest news: [www.mathworks.com](http://www.mathworks.com)  
Sales and services: [www.mathworks.com/sales\\_and\\_services](http://www.mathworks.com/sales_and_services)  
User community: [www.mathworks.com/matlabcentral](http://www.mathworks.com/matlabcentral)  
Technical support: [www.mathworks.com/support/contact\\_us](http://www.mathworks.com/support/contact_us)



Phone: 508-647-7000



The MathWorks, Inc.  
3 Apple Hill Drive  
Natick, MA 01760-2098

*SimMechanics™ Reference*

© COPYRIGHT 2002–2015 by The MathWorks, Inc.

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from The MathWorks, Inc.

FEDERAL ACQUISITION: This provision applies to all acquisitions of the Program and Documentation by, for, or through the federal government of the United States. By accepting delivery of the Program or Documentation, the government hereby agrees that this software or documentation qualifies as commercial computer software or commercial computer software documentation as such terms are used or defined in FAR 12.212, DFARS Part 227.72, and DFARS 252.227-7014. Accordingly, the terms and conditions of this Agreement and only those rights specified in this Agreement, shall pertain to and govern the use, modification, reproduction, release, performance, display, and disclosure of the Program and Documentation by the federal government (or other entity acquiring for or through the federal government) and shall supersede any conflicting contractual terms or conditions. If this License fails to meet the government's needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to The MathWorks, Inc.

### Trademarks

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See [www.mathworks.com/trademarks](http://www.mathworks.com/trademarks) for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

### Patents

MathWorks products are protected by one or more U.S. patents. Please see [www.mathworks.com/patents](http://www.mathworks.com/patents) for more information.

### Revision History

March 2012	Online only	New for Version 4.0 (Release R2012a)
September 2012	Online only	Revised for Version 4.1 (Release R2012b)
March 2013	Online only	Revised for Version 4.2 (Release R2013a)
September 2013	Online only	Revised for Version 4.3 (Release R2013b)
March 2014	Online only	Revised for Version 4.4 (Release R2014a)
October 2014	Online only	Revised for Version 4.5 (Release R2014b)
March 2015	Online only	Revised for Version 4.6 (Release R2015a)

## Blocks—Alphabetical List

1

## Configuration Parameters

2

<b>SimMechanics Pane: General</b> .....	2-2
SimMechanics Pane Overview .....	2-2
<b>SimMechanics Pane: Diagnostics</b> .....	2-3
Invalid visual properties .....	2-4
Repeated vertices in a cross-section .....	2-5
Unconnected frame port .....	2-6
Redundant block .....	2-7
Conflicting reference frames .....	2-8
Rigidly constrained block .....	2-9
Unsatisfied high priority state targets .....	2-10
Overspecified targets in kinematic loops .....	2-11
<b>SimMechanics Pane: Explorer</b> .....	2-12
Open Mechanics Explorer on model update or simulation ..	2-12

## SimMechanics Visualization

3

<b>Visualization with Mechanics Explorer</b> .....	3-2
Introduction to Mechanics Explorer .....	3-2
Visualization Requirements .....	3-4
Configure Mechanics Explorer for Automatic Start-Up .....	3-4

Open Mechanics Explorer .....	3-4
<b>Menu and Tool Bars</b> .....	<b>3-8</b>
Menu Bar .....	3-8
Toolbar .....	3-9
<b>Tree View and Property Panes</b> .....	<b>3-11</b>
Introduction to Tree View and Property Panes .....	3-11
Browse Model .....	3-11
<b>Context Menu Display Controls</b> .....	<b>3-13</b>
<b>Animation Toolstrip</b> .....	<b>3-14</b>
Requirements for Animation Recording and Playback .....	3-14
How Animation and Playback Work .....	3-14
Play Control Buttons .....	3-15
Slider and Loop Controls .....	3-15
Playback Speed Control .....	3-16
Playback Time Counter .....	3-16

## Functions—Alphabetical List

# 4

# Blocks—Alphabetical List

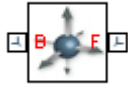
---

## 6-DOF Joint

Joint with one spherical and three prismatic primitives

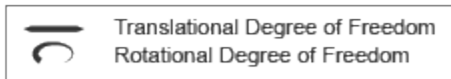
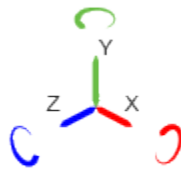
### Library

Joints



### Description

This block represents a joint with three translational and three rotational degrees of freedom. Three prismatic primitives provide the translational degrees of freedom. One spherical primitive provides the three rotational degrees of freedom.



### Joint Degrees of Freedom

The joint block represents motion between the base and follower frames as a sequence of time-varying transformations. Each joint primitive applies one transformation in this sequence. The transformation translates or rotates the follower frame with respect to the joint primitive base frame. For all but the first joint primitive, the base frame coincides with the follower frame of the previous joint primitive in the sequence.

At each time step during the simulation, the joint block applies the sequence of time-varying frame transformations in this order:

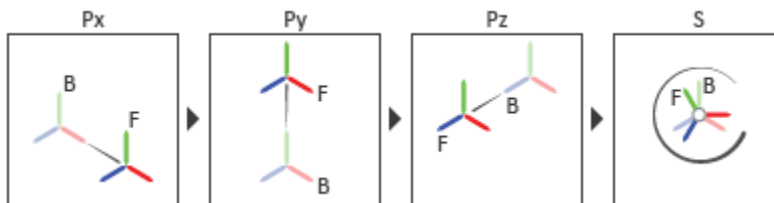
## 1 Translation:

- a Along the X axis of the X Prismatic Primitive (Px) base frame.
- b Along the Y axis of the Y Prismatic Primitive (Py) base frame. This frame is coincident with the X Prismatic Primitive (Px) follower frame.
- c Along the Z axis of the Z Prismatic Primitive (Pz) base frame. This frame is coincident with the Y Prismatic Primitive (Py) follower frame.

## 2 Rotation:

- About a general 3-D axis resolved in the base frame. This frame is coincident with the Z Prismatic Primitive (Pz) follower frame.

The figure shows the sequence in which the joint transformations occur at a given simulation time step. The resulting frame of each transformation serves as the base frame for the following transformation. Because 3-D rotation occurs as a single rotation about an arbitrary 3-D axis (as opposed to three separate rotations about the X, Y, Z axes), gimbal lock does not occur.



### Joint Transformation Sequence

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs

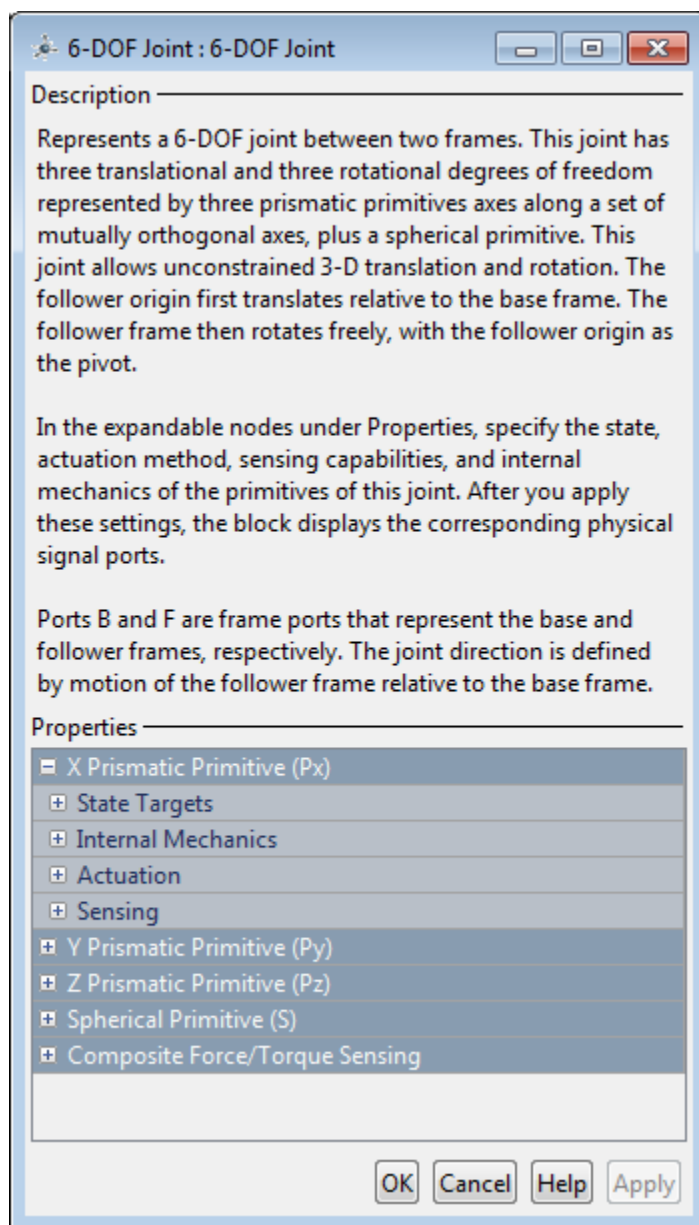
that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types: Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.





## Prismatic Primitive: State Targets

Specify the prismatic primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative position, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative velocity, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is m for position and m/s for velocity.

## Prismatic Primitive: Internal Mechanics

Specify the prismatic primitive internal mechanics. Internal mechanics include linear spring forces, accounting for energy storage, and damping forces, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

### Equilibrium Position

Enter the spring equilibrium position. This is the distance between base and follower frame origins at which the spring force is zero. The default value is 0. Select or enter a physical unit. The default is m.

### Spring Stiffness

Enter the linear spring constant. This is the force required to displace the joint primitive by a unit distance. The default is 0. Select or enter a physical unit. The default is N/m.

### Damping Coefficient

Enter the linear damping coefficient. This is the force required to maintain a constant joint primitive velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is N / (m/s).

## Prismatic Primitive: Actuation

Specify actuation options for the prismatic joint primitive. Actuation modes include **Force** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Actuation signals are resolved in the base frame.

### Force

Select an actuation force setting. The default setting is **None**.

Actuation Force Setting	Description
None	No actuation force.
Provided by Input	Actuation force from physical signal input. The signal provides the force acting on the follower frame with respect to the base frame along the joint primitive axis. An equal and opposite force acts on the base frame.

Actuation Force Setting	Description
Automatically computed	Actuation force from automatic calculation. SimMechanics™ computes and applies the actuation force based on model dynamics.

**Motion**

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

**Prismatic Primitive: Sensing**

Select the variables to sense in the prismatic joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

**Position**

Select this option to sense the relative position of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Velocity**

Select this option to sense the relative velocity of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Acceleration**

Select this option to sense the relative acceleration of the follower frame origin with respect to the base frame origin along the joint primitive axis.

### Actuator Force

Select this option to sense the actuation force acting on the follower frame with respect to the base frame along the joint primitive axis.

## Spherical Primitive: State Targets

Specify the spherical primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative rotation of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Select a method to specify the joint primitive state target.

Method	Description
None	Align base and follower frames.
Aligned Axes	Align two follower frame axes with two base frame axes.

Method	Description
Standard Axis	Specify a rotation angle about a standard axis ( $x$ , $y$ , or $z$ ).
Arbitrary Axis	Specify a rotation angle about a general [ $x$ , $y$ , $z$ ] axis.
Rotation Matrix	Specify a right-handed orthogonal rotation matrix.

### Aligned Axes

Select two pairs of base-follower frame axes.

Parameter	Description
Pair 1	First pair of base-follower frame axes to align.
Pair 2	Second pair of base-follower frame axes to align. Axis choices depend on <b>Pair 1</b> axis selections.

### Standard Axis

Select a standard rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
Axis	Standard rotation axis (X, Y, or Z) resolved in the base frame.
Angle	Follower frame rotation angle about the rotation axis with respect to the base frame.

### Arbitrary Axis

Select a general 3-D rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
Axis	General rotation axis [X Y Z] resolved in the base frame.

Parameter	Description
Angle	Follower frame rotation angle about the rotation axis with respect to the base frame.

### Rotation Matrix

Specify the 3×3 transformation matrix of a proper rotation between the base and follower frames. The matrix must be orthogonal and have determinant +1. The default value is [1 0 0; 0 1 0; 0 0 1].

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative angular velocity of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields. The priority field is identical to that used for the position state target.

### Value

Enter a three element vector with the angular velocity components of the follower frame with respect to the base frame. Select a physical unit. The default is deg/s (degree/second).

### Resolution Frame

From the drop-down list, select a resolution frame. This is the frame in which the state target is resolved. It identifies the axes the velocity vector components apply to. The default is Follower.

## Spherical Primitive: Internal Mechanics

Specify the spherical primitive internal mechanics. This includes linear spring and damping forces, accounting for energy storage and dissipation, respectively. To ignore internal mechanics, keep spring stiffness and damping coefficient values at the default value of 0.

### Equilibrium Position

Select a method to specify the spring equilibrium position. The equilibrium position is the rotation angle between base and follower port frames at which the spring torque is zero.

Method	Description
None	Align base and follower frames.

Method	Description
Aligned Axes	Align two follower frame axes with two base frame axes.
Standard Axis	Specify a rotation angle about a standard axis ( $x$ , $y$ , or $z$ ).
Arbitrary Axis	Specify a rotation angle about a general $[x, y, z]$ axis.
Rotation Matrix	Specify a right-handed orthogonal rotation matrix.

### Aligned Axes

Select two pairs of base-follower frame axes.

Parameter	Description
Pair 1	First pair of base-follower frame axes to align.
Pair 2	Second pair of base-follower frame axes to align. Axis choices depend on <b>Pair 1</b> axis selections.

### Standard Axis

Select a standard rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
Axis	Standard rotation axis (X, Y, or Z) resolved in the base frame.
Angle	Follower frame rotation angle about the rotation axis with respect to the base frame.

### Arbitrary Axis

Select a general 3-D rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
Axis	General rotation axis $[X Y Z]$ resolved in the base frame.



Parameter	Description
Angle	Follower frame rotation angle about the rotation axis with respect to the base frame.

### Rotation Matrix

Specify the 3×3 transformation matrix of a proper rotation between the base and follower frames. The matrix must be orthogonal and have determinant +1. The default value is [1 0 0; 0 1 0; 0 0 1].

### Spring Stiffness

Enter the linear spring constant. This is the torque required to displace the joint primitive by a unit angle. The term linear refers to the mathematical form of the spring equation. The default is 0. Select a physical unit. The default is N\*m/deg.

### Damping Coefficient

Enter the linear damping coefficient. This is the torque required to maintain a constant joint primitive angular velocity between base and follower frames. The default is 0. Select a physical unit. The default is N\*m/(deg/s).

## Spherical Primitive: Actuation

Specify actuation options for the spherical joint primitive. Actuation modes include **Torque** only. Selecting a torque input adds the corresponding physical signal port to the block. Use this port to specify the actuation torque signal.

### Torque

Select a source for the actuation torque. The default setting is None.

Actuation Torque Setting	Description
None	Apply no actuation torque.
Provided by Input	Apply an actuation torque based on a physical signal. The signal specifies the torque acting on the follower frame with respect to the base frame. An equal and opposite torque acts on the base frame. Selecting this option exposes additional parameters.

**Torque (X), Torque (Y), Torque (Z)**

Select in order to actuate the spherical joint primitive about each standard Cartesian axis (X, Y, Z) separately. The block exposes the corresponding physical signal ports. Use these ports to specify the actuation torque signals. The signals must be scalar values.

**Torque (XYZ)**

Select in order to actuate the spherical joint primitive about an arbitrary axis [X Y Z]. The block exposes the corresponding physical signal port. Use this port to specify the actuation torque signal. The signal must be a 3-D vector.

**Frame**

Select the frame to resolve the actuation torque signal in. The axes of this frame establish the directions of the X, Y, and Z torque components. The default setting is Base.

**Spherical Primitive: Sensing**

Select the motion variables to sense in the spherical joint primitive. The block adds the corresponding physical signal ports. Use these ports to output the numerical values of the motion variables.

The block measures each motion variable for the follower frame with respect to the base frame. It resolves that variable in the resolution frame that you select from the **Frame** drop-down list.

Motion Variables	Description
<p><b>Position</b></p>	<p>Quaternion describing follower frame rotation with respect to base frame. The quaternion coefficients are</p> $\left[ \cos\left(\frac{\theta}{2}\right), n_x \sin\left(\frac{\theta}{2}\right), n_y \sin\left(\frac{\theta}{2}\right), n_z \sin\left(\frac{\theta}{2}\right) \right]$ <p>The measurement is the same in all measurement frames.</p>
<p><b>Velocity (X), Velocity (Y), Velocity (Z)</b></p>	<p>Angular velocity components about X, Y, and Z axes.</p>
<p><b>Velocity</b></p>	<p>3-D angular velocity vector with components about X, Y, and Z axes.</p>

Motion Variables	Description
Acceleration (X), Acceleration (Y), Acceleration (Z)	Angular acceleration components about X, Y, and Z axes.
Acceleration	3-D angular acceleration vector with components about X, Y, and Z axes.

### Frame

Select the frame to resolve the measurement in. The axes of this frame establish the directions of X, Y, and Z vector components. The default setting is **Base**.

### Composite Force/Torque Sensing

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

### Direction

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

### Resolution Frame

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

**Constraint Force**

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector  $[fc_x, fc_y, fc_z]$  acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

**Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector  $[tc_x, tc_y, tc_z]$  acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

**Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector  $[ft_x, ft_y, ft_z]$ . Selecting this option causes the block to expose physical signal port *ft*.

**Total Torque**

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.

- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port  $tt$ .

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
f	Force	Actuation input	Scalar if associated with an axis (e.g. fx). Three-element vector otherwise (e.g. f).
t	Torque	Actuation input	Scalar if associated with an axis (e.g. tx). Three-element vector otherwise (e.g. t).
p	Position	Sensing output	Scalar if associated with an axis (e.g. px). Three-element vector otherwise (e.g. p).

Port Label	Description	Type	Input/Output
v	Velocity	Sensing output	Scalar if associated with an axis (e.g. vx). Three-element vector otherwise (e.g. v).
a	Acceleration	Sensing output	Scalar if associated with an axis (e.g. ax). Three-element vector otherwise (e.g. a).
Q	Quaternion rotation	Sensing output	Four-element vector
w	Angular velocity	Sensing output	Scalar if associated with an axis (e.g. wx). Three-element vector otherwise (e.g. w).
b	Angular acceleration	Sensing output	Scalar if associated with an axis (e.g. bx). Three-element vector otherwise (e.g. b).

### See Also

Bushing Joint | Prismatic Joint | Spherical Joint

### More About

- “Motion Sensing”
- “Measurement Frames”
- “Actuating and Sensing Using Physical Signals”

# Angle Constraint

Fixed angle between two frame Z axes

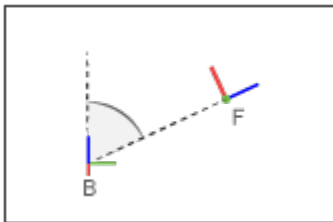
## Library

Constraints

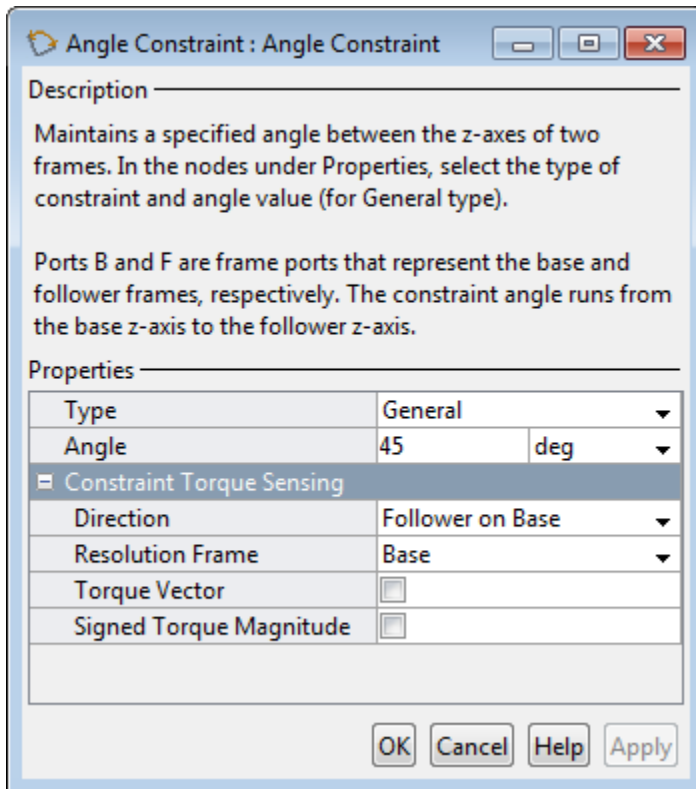


## Description

This block applies a fixed angle between the Z axes of the base and follower port frames. The frames lose one rotational degree of freedom if the constraint angle is greater than  $0^\circ$  and smaller than  $180^\circ$ . They lose two rotational degrees of freedom if the constraint angle is exactly  $0^\circ$  or  $180^\circ$ —that is, if the frames are parallel or anti-parallel. The figure shows the constraint angle between two frames.



## Dialog Box and Parameters



### Type

Angle constraint type. The default setting is General.

Type	Purpose
Parallel	Align the base and follower frame +Z axes.
Anti-Parallel	Align the base frame +Z axis with the follower frame -Z axis.
Perpendicular	Make the base and follower frame Z axes perpendicular to each other.



Type	Purpose
General	Hold the specified angle between the Z axes of the base and follower port frames.

### Angle

Constraint angle between the base and follower frame Z axes. The angle must lie in the range  $0 < \theta < 180$  deg. For an angle of 0 or 180 deg, set **Type** to **Parallel** or **Anti-Parallel** instead. The default value is 45 deg.

### Constraint Torque Sensing

Select whether to compute and output the distance constraint torque vector and its magnitude. The distance constraint torque is the torque the block must apply in order to maintain the angle you specify between the base and follower port frames.

#### Direction

Constraint torques act in pairs. As expressed by Newton's third law of motion, if the base port frame exerts a constraint torque on the follower port frame, then the follower port frame must exert an equal and opposite torque on the base port frame. Select which of the two constraint torques to sense:

- **Follower on Base** — Sense the constraint torque that the follower port frame exerts on the base port frame.
- **Base on Follower** — Sense the constraint torque that the base port frame exerts on the follower port frame.

#### Resolution Frame

The block expresses the constraint torque vector in terms of its Cartesian vector components. The splitting of a vector into vector components is known as vector resolution. The frame whose axes define the vector component directions is known as the resolution frame. Select whether to resolve the constraint torque vector in the base or follower port frame.

#### Torque Vector

Compute and output the Cartesian components of the angle constraint torque vector. The output signal is a three-dimensional vector with components expressed about the X, Y, and Z axes of the resolution frame.

#### Signed Torque Magnitude

Compute and output the magnitude of the angle constraint torque, including its sign.

## Ports

The block provides two frame ports:

- B — Base frame port
- F — Follower frame port

In addition, the block provides two physical signal output ports:

- $t$  — Angle constraint torque vector
- $tm$  — Signed magnitude of the angle constraint torque

## See Also

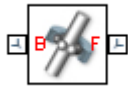
Distance Constraint

# Bearing Joint

Joint with one prismatic and three revolute primitives

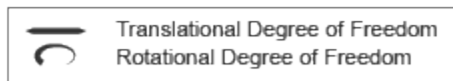
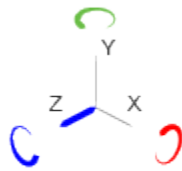
## Library

Joints



## Description

This block represents a joint with one translational and three rotational degrees of freedom. One prismatic primitive provides the translational degree of freedom. Three revolute primitives provide the three rotational degrees of freedom.



## Joint Degrees of Freedom

The joint block represents motion between the base and follower frames as a sequence of time-varying transformations. Each joint primitive applies one transformation in this sequence. The transformation translates or rotates the follower frame with respect to the joint primitive base frame. For all but the first joint primitive, the base frame coincides with the follower frame of the previous joint primitive in the sequence.

At each time step during the simulation, the joint block applies the sequence of time-varying frame transformations in this order:

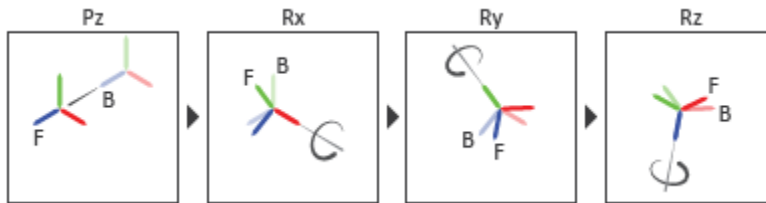
1 Translation:

- Along the Z axis of the Z Prismatic Primitive (Pz) base frame.

2 Rotation:

- a About the X axis of the X Revolute Primitive (Rx) base frame. This frame is coincident with the Z Prismatic Primitive (Pz) follower frame.
- b About the Y axis of the Y Revolute Primitive (Ry) base frame. This frame is coincident with the X Revolute Primitive (Rx) follower frame.
- c About the Z axis of the Z Revolute Primitive (Rz) base frame. This frame is coincident with the Y Revolute Primitive (Ry) follower frame.

The figure shows the sequence in which the joint transformations occur at a given simulation time step. The resulting frame of each transformation serves as the base frame for the following transformation. Because 3-D rotation occurs as a sequence, it is possible for two axes to align, causing to the loss of one rotational degree of freedom. This phenomenon is known as gimbal lock.



**Joint Transformation Sequence**

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

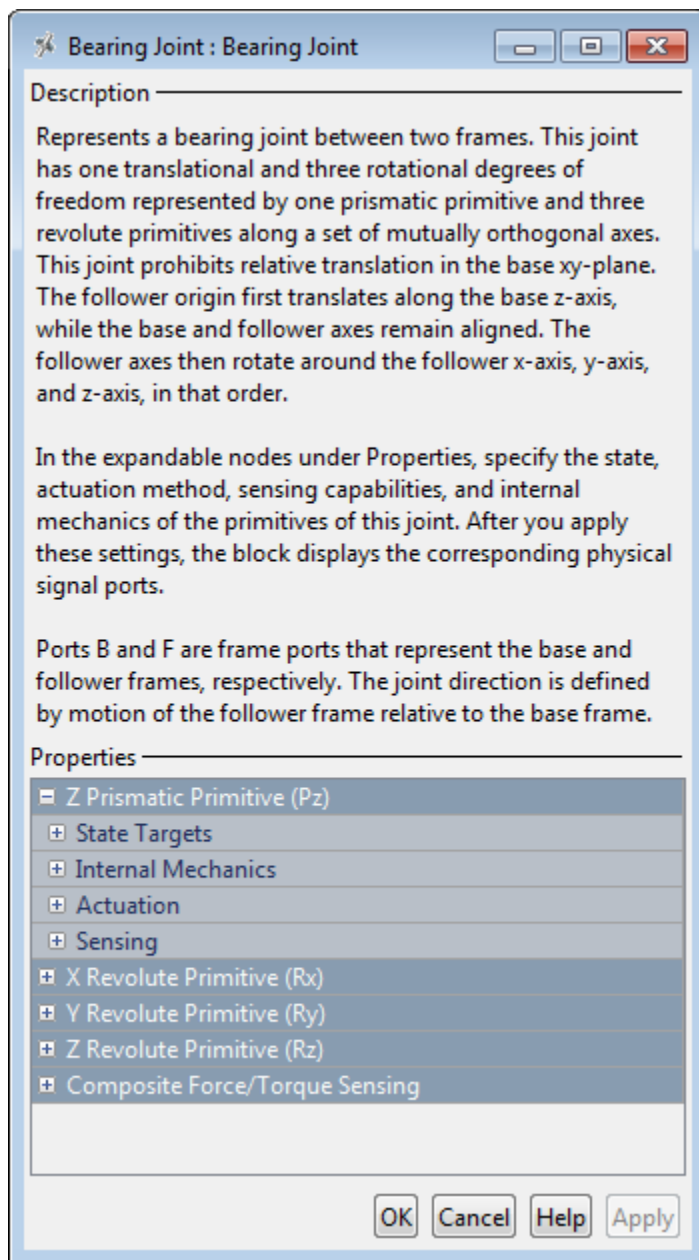
Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs

that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types: Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.



## Prismatic Primitive: State Targets

Specify the prismatic primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative position, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative velocity, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is m for position and m/s for velocity.

## Prismatic Primitive: Internal Mechanics

Specify the prismatic primitive internal mechanics. Internal mechanics include linear spring forces, accounting for energy storage, and damping forces, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

### Equilibrium Position

Enter the spring equilibrium position. This is the distance between base and follower frame origins at which the spring force is zero. The default value is 0. Select or enter a physical unit. The default is m.

### Spring Stiffness

Enter the linear spring constant. This is the force required to displace the joint primitive by a unit distance. The default is 0. Select or enter a physical unit. The default is N/m.

### Damping Coefficient

Enter the linear damping coefficient. This is the force required to maintain a constant joint primitive velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is N / (m/s).

## Prismatic Primitive: Actuation

Specify actuation options for the prismatic joint primitive. Actuation modes include **Force** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Actuation signals are resolved in the base frame.

### Force

Select an actuation force setting. The default setting is **None**.

Actuation Force Setting	Description
None	No actuation force.
Provided by Input	Actuation force from physical signal input. The signal provides the force acting on the follower frame with respect to the base frame along the joint primitive axis. An equal and opposite force acts on the base frame.



Actuation Force Setting	Description
Automatically computed	Actuation force from automatic calculation. SimMechanics computes and applies the actuation force based on model dynamics.

### Motion

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

### Prismatic Primitive: Sensing

Select the variables to sense in the prismatic joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

#### Position

Select this option to sense the relative position of the follower frame origin with respect to the base frame origin along the joint primitive axis.

#### Velocity

Select this option to sense the relative velocity of the follower frame origin with respect to the base frame origin along the joint primitive axis.

#### Acceleration

Select this option to sense the relative acceleration of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Actuator Force**

Select this option to sense the actuation force acting on the follower frame with respect to the base frame along the joint primitive axis.

**Revolute Primitive: State Targets**

Specify the revolute primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

**Specify Position Target**

Select this option to specify the desired joint primitive position at time zero. This is the relative rotation angle, measured about the joint primitive axis, of the follower frame with respect to the base frame. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

**Specify Velocity Target**

Select this option to specify the desired joint primitive velocity at time zero. This is the relative angular velocity, measured about the joint primitive axis, of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields.

**Priority**

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

**Value**

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is **deg** for position and **deg/s** for velocity.

**Revolute Primitive: Internal Mechanics**

Specify the revolute primitive internal mechanics. Internal mechanics include linear spring torques, accounting for energy storage, and linear damping torques, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

**Equilibrium Position**

Enter the spring equilibrium position. This is the rotation angle between base and follower frames at which the spring torque is zero. The default value is 0. Select or enter a physical unit. The default is **deg**.

**Spring Stiffness**

Enter the linear spring constant. This is the torque required to rotate the joint primitive by a unit angle. The default is 0. Select or enter a physical unit. The default is **N\*m/deg**.

**Damping Coefficient**

Enter the linear damping coefficient. This is the torque required to maintain a constant joint primitive angular velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is **N\*m/(deg/s)**.

**Revolute Primitive: Actuation**

Specify actuation options for the revolute joint primitive. Actuation modes include **Torque** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Input signals are resolved in the base frame.

**Torque**

Select an actuation torque setting. The default setting is **None**.

Actuation Torque Setting	Description
None	No actuation torque.

Actuation Torque Setting	Description
Provided by Input	Actuation torque from physical signal input. The signal provides the torque acting on the follower frame with respect to the base frame about the joint primitive axis. An equal and opposite torque acts on the base frame.
Automatically computed	Actuation torque from automatic calculation. SimMechanics computes and applies the actuation torque based on model dynamics.

### Motion

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

### Revolute Primitive: Sensing

Select the variables to sense in the revolute joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

### Position

Select this option to sense the relative rotation angle of the follower frame with respect to the base frame about the joint primitive axis.

**Velocity**

Select this option to sense the relative angular velocity of the follower frame with respect to the base frame about the joint primitive axis.

**Acceleration**

Select this option to sense the relative angular acceleration of the follower frame with respect to the base frame about the joint primitive axis.

**Actuator Torque**

Select this option to sense the actuation torque acting on the follower frame with respect to the base frame about the joint primitive axis.

**Composite Force/Torque Sensing**

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

**Direction**

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

**Resolution Frame**

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

**Constraint Force**

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector  $[fc_x, fc_y, fc_z]$  acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

**Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector  $[tc_x, tc_y, tc_z]$  acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

**Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector  $[ft_x, ft_y, ft_z]$ . Selecting this option causes the block to expose physical signal port *ft*.

**Total Torque**

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.

- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port  $tt$ .

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
f	Force	Actuation input	Scalar
t	Torque	Actuation input	Scalar
p	Position	Sensing output	Scalar
v	Velocity	Sensing output	Scalar
a	Acceleration	Sensing output	Scalar
q	Rotation angle	Sensing output	Scalar
w	Angular velocity	Sensing output	Scalar
b	Angular acceleration	Sensing output	Scalar

## See Also

Prismatic Joint | Revolute Joint

## **More About**

- “Actuating and Sensing Using Physical Signals”
- “Motion Sensing”
- “Rotational Measurements”
- “Translational Measurements”

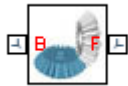


# Bevel Gear Constraint

Kinematic constraint for transferring rotational motion at a fixed ratio about arbitrarily oriented axes

## Library

Gears and Couplings/Gears



## Description

This block represents a bevel gear constraint between two frames. The constraint restricts motion such that rotation of the base frame causes rotation of the follower frame and vice-versa. The two frames spin as meshed gears about the individual Z axes, each aligned with a bevel gear shaft.

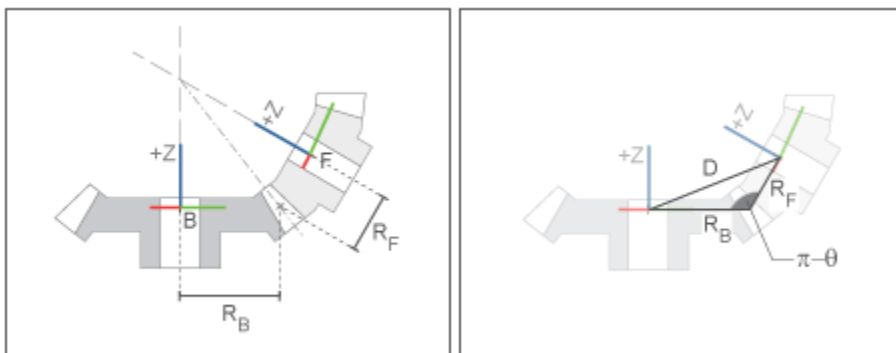
The two shafts can lie at an arbitrary angle to each other. This angle, always between  $0^\circ$  and  $180^\circ$ , corresponds to a configuration in which the two +Z axes are either parallel or converging toward each other. At  $90^\circ$  the two shafts are perpendicular. At  $0^\circ$ , the two gears are parallel and the bevel gear functions as a spur gear.

During simulation, the remainder of the mechanism must hold the gears in alignment. The gear shafts must lie at the distance and angle specified in the block dialog box. The distance between the base and follower frame origins, which represent the two gear shafts, follows from the law of cosines:

$$D^2 = R_B^2 + R_F^2 - 2 \cdot R_B \cdot R_F \cdot \cos(\pi - \theta),$$

where:

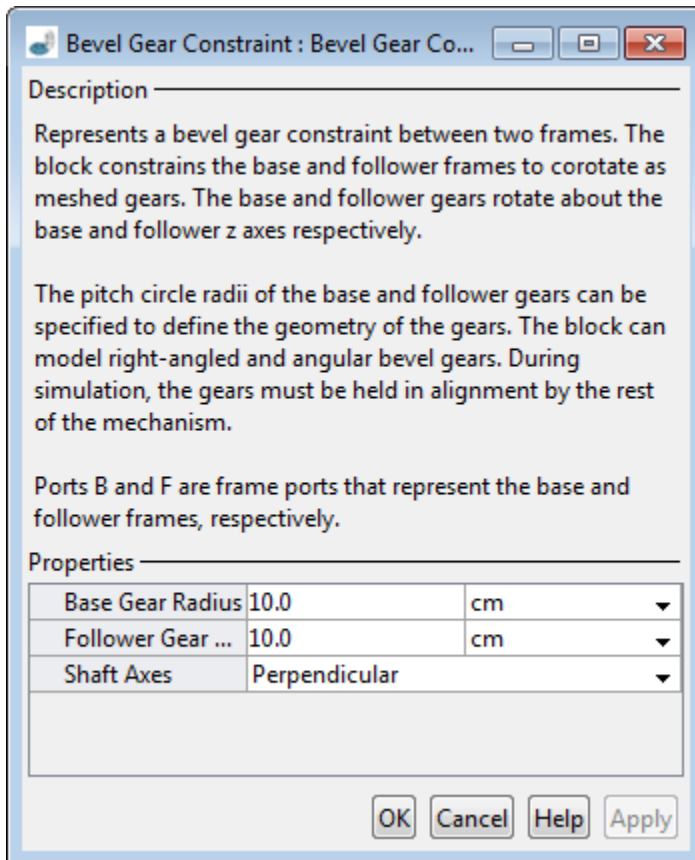
- $D$  is the distance between the two frame origins.
- $R_B, R_F$  are the pitch radii of the base gear and follower gears.
- $\theta$  is the angle between the base and follower gear shafts.



**Bevel gear Schematic**

## Dialog Box and Parameters

The block dialog box contains a **Properties** area with bevel gear options and parameters.



### Base Gear Radius

Enter the pitch circle radius of the base gear. This is the radius at which the base gear teeth contact the follower gear teeth. Select a physical unit. The default is 10.0 cm.

### Follower Gear Radius

Enter the pitch circle radius of the follower gear. This is the radius at which the follower gear teeth contact the base gear teeth. Select a physical unit. The default is 10.0 cm.

### Shaft Axes

Select the relative orientation between the spinning shafts of the base and follower gears. Options include **Perpendicular** and **Arbitrarily Oriented**:

- **Perpendicular** — Orient the two spinning shafts at a right angle to each other. This is the default setting.
- **Arbitrarily Oriented** — Orient the two spinning shafts at a general angle to each other. Selecting this option exposes an additional parameter.

**Angle Between Shafts**

Enter the angle between the spinning shafts of the base and follower gears. Select a physical unit. The default value is **90** deg.

## **Ports**

The block contains frame ports B and F, representing base and follower frames, respectively.

## **See Also**

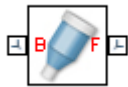
Rack and Pinion Constraint | Common Gear Constraint

# Bushing Joint

Joint with three prismatic and three revolute primitives

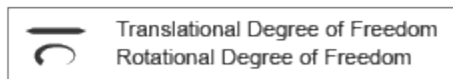
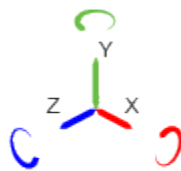
## Library

Joints



## Description

This block represents a joint with three translational and three rotational degrees of freedom. Three prismatic primitives provide the translational degrees of freedom. Three revolute primitives provide the rotational degrees of freedom.



## Joint Degrees of Freedom

The joint block represents motion between the base and follower frames as a sequence of time-varying transformations. Each joint primitive applies one transformation in this sequence. The transformation translates or rotates the follower frame with respect to the joint primitive base frame. For all but the first joint primitive, the base frame coincides with the follower frame of the previous joint primitive in the sequence.

At each time step during the simulation, the joint block applies the sequence of time-varying frame transformations in this order:

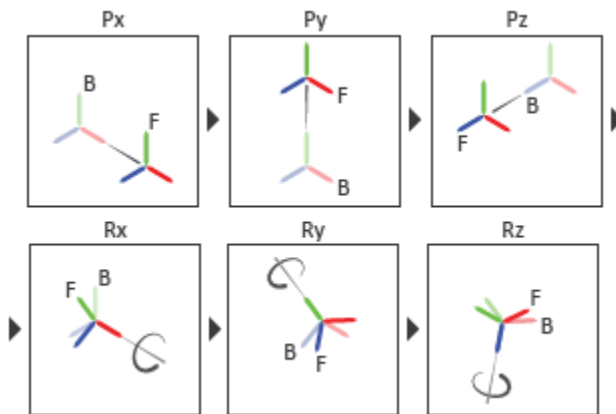
1 Translation:

- a Along the X axis of the X Prismatic Primitive (Px) base frame.
- b Along the Y axis of the Y Prismatic Primitive (Py) base frame. This frame is coincident with the X Prismatic Primitive (Px) follower frame.
- c Along the Z axis of the Z Prismatic Primitive (Pz) base frame. This frame is coincident with the Y Prismatic Primitive (Py) follower frame.

2 Rotation:

- a About the X axis of the X Revolute Primitive (Rx) base frame. This frame is coincident with the Z Prismatic Primitive (Pz) follower frame.
- b About the Y axis of the Y Revolute Primitive (Ry) base frame. This frame is coincident with the X Revolute Primitive (Rx) follower frame.
- c About the Z axis of the Z Revolute Primitive (Rz) base frame. This frame is coincident with the Y Revolute Primitive (Ry) follower frame.

The figure shows the sequence in which the joint transformations occur at a given simulation time step. The resulting frame of each transformation serves as the base frame for the following transformation. Because 3-D rotation occurs as a sequence, it is possible for two axes to align, causing to the loss of one rotational degree of freedom. This phenomenon is known as gimbal lock.



**Joint Transformation Sequence**

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

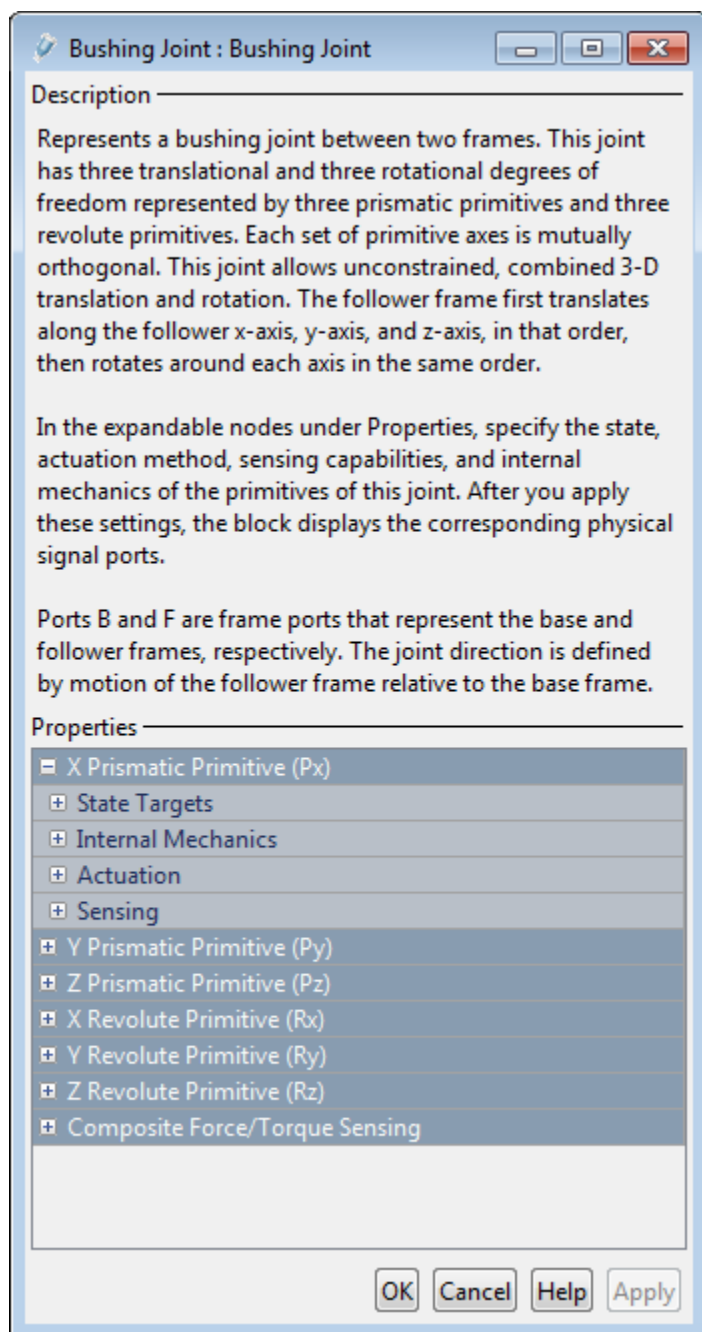
Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types: Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.





## Prismatic Primitive: State Targets

Specify the prismatic primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative position, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative velocity, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is m for position and m/s for velocity.

## Prismatic Primitive: Internal Mechanics

Specify the prismatic primitive internal mechanics. Internal mechanics include linear spring forces, accounting for energy storage, and damping forces, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

### Equilibrium Position

Enter the spring equilibrium position. This is the distance between base and follower frame origins at which the spring force is zero. The default value is 0. Select or enter a physical unit. The default is m.

### Spring Stiffness

Enter the linear spring constant. This is the force required to displace the joint primitive by a unit distance. The default is 0. Select or enter a physical unit. The default is N/m.

### Damping Coefficient

Enter the linear damping coefficient. This is the force required to maintain a constant joint primitive velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is N / (m/s).

## Prismatic Primitive: Actuation

Specify actuation options for the prismatic joint primitive. Actuation modes include **Force** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Actuation signals are resolved in the base frame.

### Force

Select an actuation force setting. The default setting is **None**.

Actuation Force Setting	Description
None	No actuation force.
Provided by Input	Actuation force from physical signal input. The signal provides the force acting on the follower frame with respect to the base frame along the joint primitive axis. An equal and opposite force acts on the base frame.

Actuation Force Setting	Description
Automatically computed	Actuation force from automatic calculation. SimMechanics computes and applies the actuation force based on model dynamics.

### Motion

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

### Prismatic Primitive: Sensing

Select the variables to sense in the prismatic joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

#### Position

Select this option to sense the relative position of the follower frame origin with respect to the base frame origin along the joint primitive axis.

#### Velocity

Select this option to sense the relative velocity of the follower frame origin with respect to the base frame origin along the joint primitive axis.

#### Acceleration

Select this option to sense the relative acceleration of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Actuator Force**

Select this option to sense the actuation force acting on the follower frame with respect to the base frame along the joint primitive axis.

**Revolute Primitive: State Targets**

Specify the revolute primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

**Specify Position Target**

Select this option to specify the desired joint primitive position at time zero. This is the relative rotation angle, measured about the joint primitive axis, of the follower frame with respect to the base frame. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

**Specify Velocity Target**

Select this option to specify the desired joint primitive velocity at time zero. This is the relative angular velocity, measured about the joint primitive axis, of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields.

**Priority**

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

**Value**

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is **deg** for position and **deg/s** for velocity.

**Revolute Primitive: Internal Mechanics**

Specify the revolute primitive internal mechanics. Internal mechanics include linear spring torques, accounting for energy storage, and linear damping torques, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

**Equilibrium Position**

Enter the spring equilibrium position. This is the rotation angle between base and follower frames at which the spring torque is zero. The default value is 0. Select or enter a physical unit. The default is **deg**.

**Spring Stiffness**

Enter the linear spring constant. This is the torque required to rotate the joint primitive by a unit angle. The default is 0. Select or enter a physical unit. The default is **N\*m/deg**.

**Damping Coefficient**

Enter the linear damping coefficient. This is the torque required to maintain a constant joint primitive angular velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is **N\*m/(deg/s)**.

**Revolute Primitive: Actuation**

Specify actuation options for the revolute joint primitive. Actuation modes include **Torque** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Input signals are resolved in the base frame.

**Torque**

Select an actuation torque setting. The default setting is **None**.

Actuation Torque Setting	Description
None	No actuation torque.

Actuation Torque Setting	Description
Provided by Input	Actuation torque from physical signal input. The signal provides the torque acting on the follower frame with respect to the base frame about the joint primitive axis. An equal and opposite torque acts on the base frame.
Automatically computed	Actuation torque from automatic calculation. SimMechanics computes and applies the actuation torque based on model dynamics.

### Motion

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

### Revolute Primitive: Sensing

Select the variables to sense in the revolute joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

### Position

Select this option to sense the relative rotation angle of the follower frame with respect to the base frame about the joint primitive axis.

**Velocity**

Select this option to sense the relative angular velocity of the follower frame with respect to the base frame about the joint primitive axis.

**Acceleration**

Select this option to sense the relative angular acceleration of the follower frame with respect to the base frame about the joint primitive axis.

**Actuator Torque**

Select this option to sense the actuation torque acting on the follower frame with respect to the base frame about the joint primitive axis.

**Composite Force/Torque Sensing**

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

**Direction**

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

**Resolution Frame**

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

**Constraint Force**

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector  $[fc_x, fc_y, fc_z]$  acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

**Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector  $[tc_x, tc_y, tc_z]$  acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

**Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector  $[ft_x, ft_y, ft_z]$ . Selecting this option causes the block to expose physical signal port *ft*.

**Total Torque**

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.



- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port  $tt$ .

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
f	Force	Actuation input	Scalar
t	Torque	Actuation input	Scalar
p	Position	Sensing output	Scalar
v	Velocity	Sensing output	Scalar
a	Acceleration	Sensing output	Scalar
q	Rotation angle	Sensing output	Scalar
w	Angular velocity	Sensing output	Scalar
b	Angular acceleration	Sensing output	Scalar

## See Also

6-DOF Joint | Gimbal Joint | Prismatic Joint | Revolute Joint

## **More About**

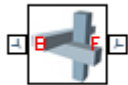
- “Actuating and Sensing Using Physical Signals”
- “Motion Sensing”
- “Rotational Measurements”
- “Translational Measurements”

# Cartesian Joint

Joint with three prismatic primitives

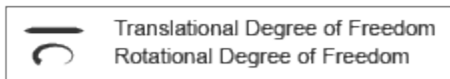
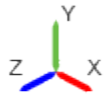
## Library

Joints



## Description

This block represents a joint with three translational degrees of freedom. Three prismatic primitives provide the three translational degrees of freedom. The base and follower frames remain parallel during simulation.



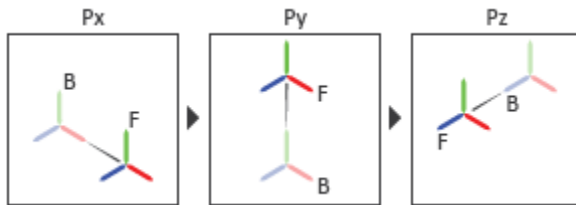
## Joint Degrees of Freedom

The joint block represents motion between the base and follower frames as a sequence of time-varying transformations. Each joint primitive applies one transformation in this sequence. The transformation translates the follower frame with respect to the joint primitive base frame. For all but the first joint primitive, the base frame coincides with the follower frame of the previous joint primitive in the sequence.

At each time step during the simulation, the joint block applies the sequence of time-varying frame transformations in this order:

- 1 Translation:
  - a Along the X axis of the X Prismatic Primitive (Px) base frame.
  - b Along the Y axis of the Y Prismatic Primitive (Py) base frame. This frame is coincident with the X Prismatic Primitive (Px) follower frame.
  - c Along the Z axis of the Z Prismatic Primitive (Pz) base frame. This frame is coincident with the Y Prismatic Primitive (Py) follower frame.

The figure shows the sequence in which the joint transformations occur at a given simulation time step. The resulting frame of each transformation serves as the base frame for the following transformation.



### Joint Transformation Sequence

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

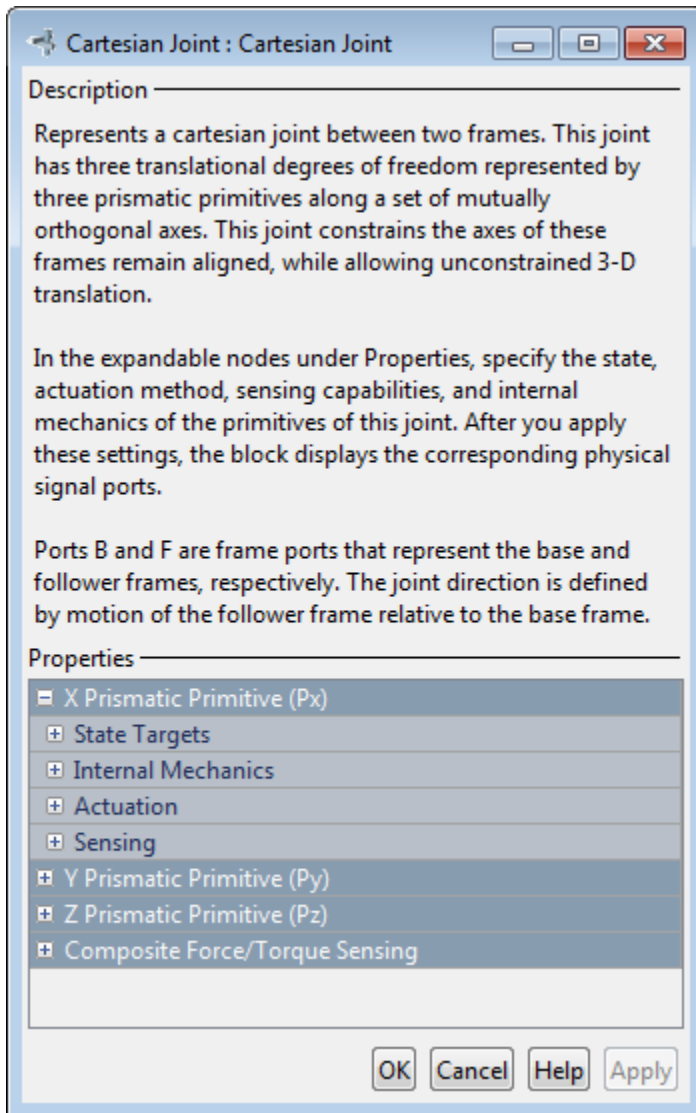
Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types: Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.



## Prismatic Primitive: State Targets

Specify the prismatic primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The

priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative position, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative velocity, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is m for position and m/s for velocity.

## Prismatic Primitive: Internal Mechanics

Specify the prismatic primitive internal mechanics. Internal mechanics include linear spring forces, accounting for energy storage, and damping forces, accounting for energy

dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

**Equilibrium Position**

Enter the spring equilibrium position. This is the distance between base and follower frame origins at which the spring force is zero. The default value is 0. Select or enter a physical unit. The default is m.

**Spring Stiffness**

Enter the linear spring constant. This is the force required to displace the joint primitive by a unit distance. The default is 0. Select or enter a physical unit. The default is N/m.

**Damping Coefficient**

Enter the linear damping coefficient. This is the force required to maintain a constant joint primitive velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is N / (m/s).

**Prismatic Primitive: Actuation**

Specify actuation options for the prismatic joint primitive. Actuation modes include **Force** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Actuation signals are resolved in the base frame.

**Force**

Select an actuation force setting. The default setting is **None**.

Actuation Force Setting	Description
None	No actuation force.
Provided by Input	Actuation force from physical signal input. The signal provides the force acting on the follower frame with respect to the base frame along the joint primitive axis. An equal and opposite force acts on the base frame.
Automatically computed	Actuation force from automatic calculation. SimMechanics computes and applies the actuation force based on model dynamics.



**Motion**

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

**Prismatic Primitive: Sensing**

Select the variables to sense in the prismatic joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

**Position**

Select this option to sense the relative position of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Velocity**

Select this option to sense the relative velocity of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Acceleration**

Select this option to sense the relative acceleration of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Actuator Force**

Select this option to sense the actuation force acting on the follower frame with respect to the base frame along the joint primitive axis.

## Composite Force/Torque Sensing

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

### Direction

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

### Resolution Frame

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

### Constraint Force

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector  $[f_{c_x}, f_{c_y}, f_{c_z}]$  acting at the joint. Only constraint force components that are orthogonal to the

joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

### Constraint Torque

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector  $[tc_x, tc_y, tc_z]$  acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

### Total Force

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector  $[ft_x, ft_y, ft_z]$ . Selecting this option causes the block to expose physical signal port *ft*.

### Total Torque

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.
- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port *tt*.

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
f	Force	Actuation input	Scalar
p	Position	Sensing output	Scalar
v	Velocity	Sensing output	Scalar
a	Acceleration	Sensing output	Scalar

## See Also

Prismatic Joint | Rectangular Joint

## More About

- “Actuating and Sensing Using Physical Signals”
- “Motion Sensing”
- “Translational Measurements”

# Common Gear Constraint

Kinematic constraint for transferring rotational motion at a fixed ratio about parallel axes

## Library

Gears and Couplings/Gears



## Description

This block represents a common gear constraint between two frames, base and follower. The constraint restricts motion, forcing the frames to spin in sync. Each frame, possessing at least one rotational degree of freedom, spins about its individual Z axis.

To assemble, the common gear constraint imposes several restrictions on the relative position and orientation of the port frames. Through blocks such as Rigid Transform, Joints, and Constraints, the remainder of the model must provide the geometric and kinematic relationships to satisfy these restrictions:

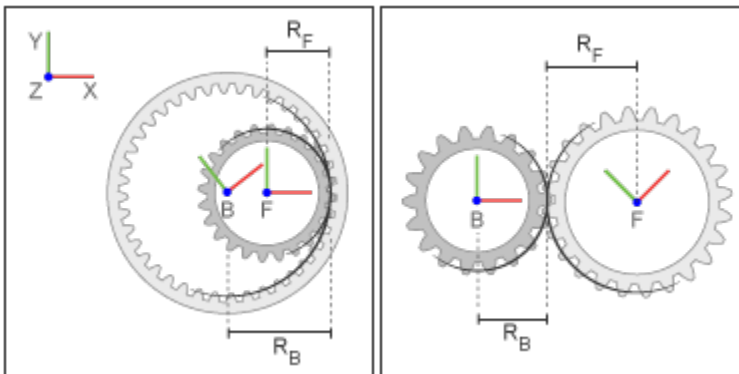
- The Z axes of the base and follower frames must point in the same direction.
- The X and Y axes of the base and follower frames, and therefore the frame origins themselves, must lie in the same plane.

The common gear constraint can be internal or external. If the constraint is internal, the base and follower gear shafts spin in the same direction. If it is external, the shafts spin in opposite directions. The tooth ratio between the two gears determines the magnitude of the angular velocity ratio:

$$\left| \frac{\omega_F}{\omega_B} \right| = \frac{N_B}{N_F} = \frac{R_B}{R_F},$$

where:

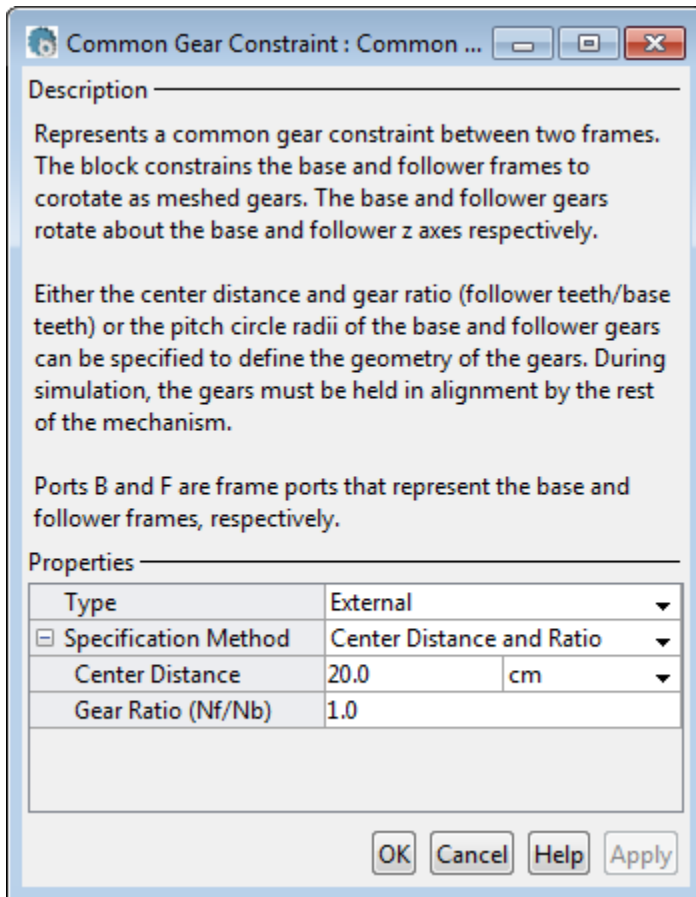
- $\omega_B$  and  $\omega_F$  are the angular velocities of the base and follower gears.
- $N_B$  and  $N_F$  are the tooth numbers of the base and follower gears.
- $R_B$  and  $R_F$  are the pitch radii of the base and follower gears. These are the radii of the gear pitch circles, imaginary circles tangent to each other at the tooth-tooth contact point.



### Gear Frames and Dimensions

## Dialog Box and Parameters

The block dialog box contains a **Properties** area with common gear options and parameters.



### Type

Select gear type. Options include internal or external. Internal gears have teeth along the inner circumference. External gears have teeth along the outer circumference. The default type is **External**.

### Specification Method

Select the method to specify the common gear with. Options include **Center Distance and Ratio** and **Pitch Circle Radii**.

## Center Distance and Ratio

Specify the common gear constraint in terms of the center-to-center separation distance and the gear teeth ratio ( $N_f/N_b$ ).

### Center Distance

Enter the center-to-center distance between base and follower gears. Select a physical unit. The default value is 20.0 cm.

### Gear Ratio

Enter the gear teeth ratio between base and follower gears. This ratio is  $N_f/N_b$ , where  $N_f$  and  $N_b$  represent the number of teeth in the follower and base gears, respectively. The default value is 1.0.

## Pitch Circle Radii

Specify the common gear in terms of the base and follower gear radii.

### Base Gear Radius

Enter the radius of the gear associated with the base frame. Select a physical unit. The default value is 10.0 cm.

### Follower Gear Radius

Enter the radius of the gear associated with the follower frame. Select a physical unit. The default value is 10.0 cm.

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively.

## See Also

Rack and Pinion Constraint | Bevel Gear Constraint

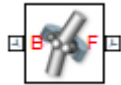


# Constant Velocity Joint

Joint with two rotational DoFs between shafts constrained to spin with equal velocity

## Library

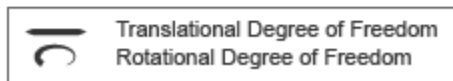
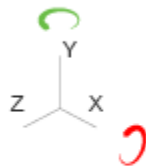
Joints



## Description

This block represents a joint with two rotational degrees of freedom constrained to maintain a constant angular velocity about the base and follower Z axes. The base and follower frame origins remain coincident throughout simulation.

The joint applies three rotation transformations between the base and follower frames in the sequence azimuth  $\rightarrow$  bend angle  $\rightarrow$  -azimuth. Each transformation takes place relative to the intermediate frame resulting from any prior transformations. For example, the bend angle transformation takes place relative to the intermediate frame resulting from the azimuth transformation.

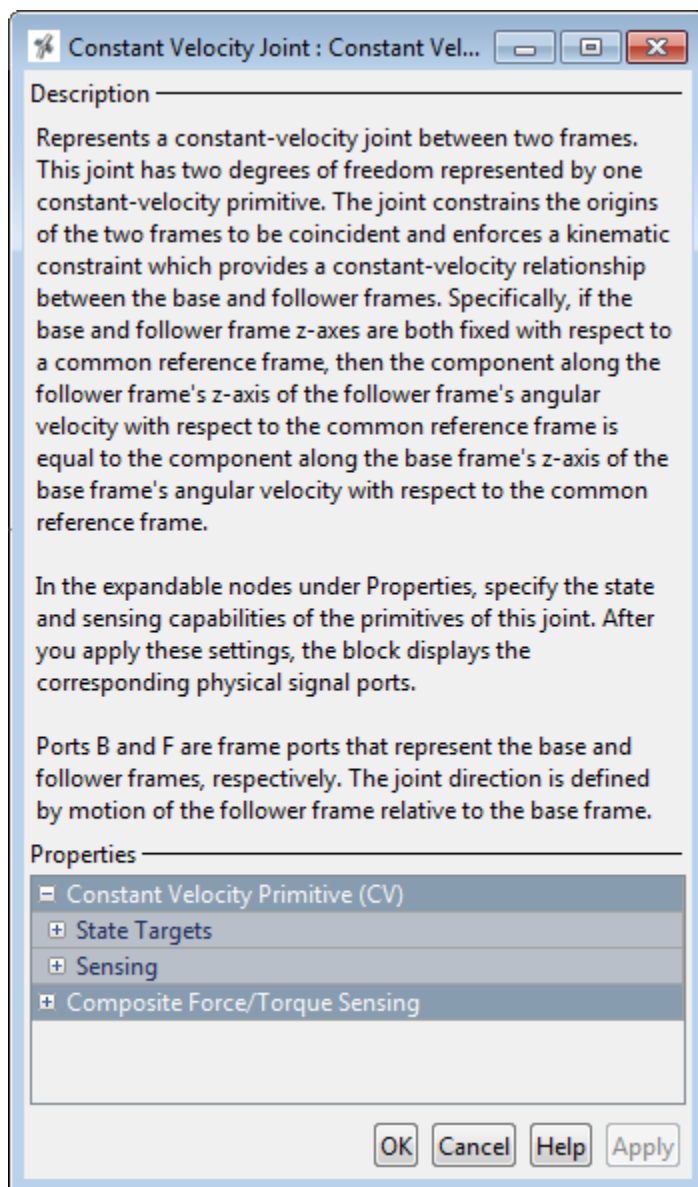


## Joint Degrees of Freedom

A set of optional state targets guide assembly for the joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

Optional sensing ports output the joint primitive motion through physical signals. Motion variables that you can sense include joint position, velocity, and acceleration. Selecting a variable in the Sensing menu exposes the physical signal port for that variable.

## Dialog Box and Parameters



## Constant Velocity Primitive: State Targets

### Specify Position Target

Desired joint primitive position at the start of simulation. This is the relative angular position of the follower frame relative to the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Desired joint velocity at the start of simulation. This is the relative angular velocity of the follower frame relative to the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Joint primitive angles to specify. Angles include bend and azimuth angles.

#### Value: Bend Angle

Angle between the base and follower frame Z axes. The block applies this angle about the rotated Y axis resulting from the azimuth transformation. At zero bend angle, the follower frame Z axis is coincident with the base frame Z axis.

#### Value: Azimuth

Angle about the base frame Z axis prior to bending. At zero azimuth, the base and follower Z axes are in the XZ plane of the base frame.

## Constant Velocity Primitive: Sensing

Select the variables to sense in the constant velocity joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame.

Variable	Description
Bend Angle: <b>Position</b>	Angle between the base and follower frame Z axes
Bend Angle: <b>Velocity</b>	First time derivative of the bend angle.
Bend Angle: <b>Acceleration</b>	Second time derivative of the bend angle.
Azimuth: <b>Position</b>	Angle about the base frame Z axis prior to bending.
Azimuth: <b>Velocity</b>	First time derivative of the azimuth angle.
Azimuth: <b>Acceleration</b>	Second time derivative of the azimuth angle.

## Composite Force/Torque Sensing

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

### Direction

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are

positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

### **Resolution Frame**

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

### **Constraint Force**

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector [ $f_{c_x}$ ,  $f_{c_y}$ ,  $f_{c_z}$ ] acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

### **Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector [ $t_{c_x}$ ,  $t_{c_y}$ ,  $t_{c_z}$ ] acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

### **Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.

- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector  $[ft_x, ft_y, ft_z]$ . Selecting this option causes the block to expose physical signal port  $ft$ .

### Total Torque

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.
- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port  $tt$ .

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description
qb	Bend angle
wb	Bend angle first time derivative

Port Label	Description
bb	Bend angle second time derivative
qa	Azimuth
wa	Azimuth first time derivative
ba	Azimuth second time derivative
fc	Constraint force
tc	Constraint torque
ft	Total force
tt	Total torque

### See Also

Prismatic Joint | Spherical Joint

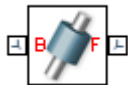


# Cylindrical Joint

Joint with one prismatic and one revolute primitives possessing parallel motion axes

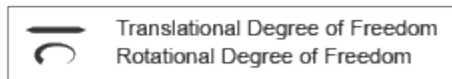
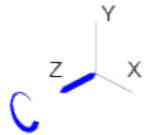
## Library

Joints



## Description

This block represents a joint with one translational and one rotational degree of freedom. One prismatic primitive provides the translational degree of freedom. One revolute primitive provides the rotational degree of freedom. The translation and rotation axes remain aligned during simulation.



## Joint Degrees of Freedom

The joint block represents motion between the base and follower frames as a sequence of time-varying transformations. Each joint primitive applies one transformation in this sequence. The transformation translates or rotates the follower frame with respect to the joint primitive base frame. For all but the first joint primitive, the base frame coincides with the follower frame of the previous joint primitive in the sequence.

At each time step during the simulation, the joint block applies the sequence of time-varying frame transformations in this order:

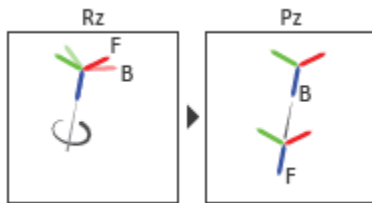
**1** Rotation:

- About the Z axis of the Z Revolute Primitive (Rz) base frame.

**2** Translation:

- Along the Z axis of the Z Prismatic Primitive (Pz) base frame.

The figure shows the sequence in which the joint transformations occur at a given simulation time step. The resulting frame of each transformation serves as the base frame for the following transformation.



### Joint Transformation Sequence

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

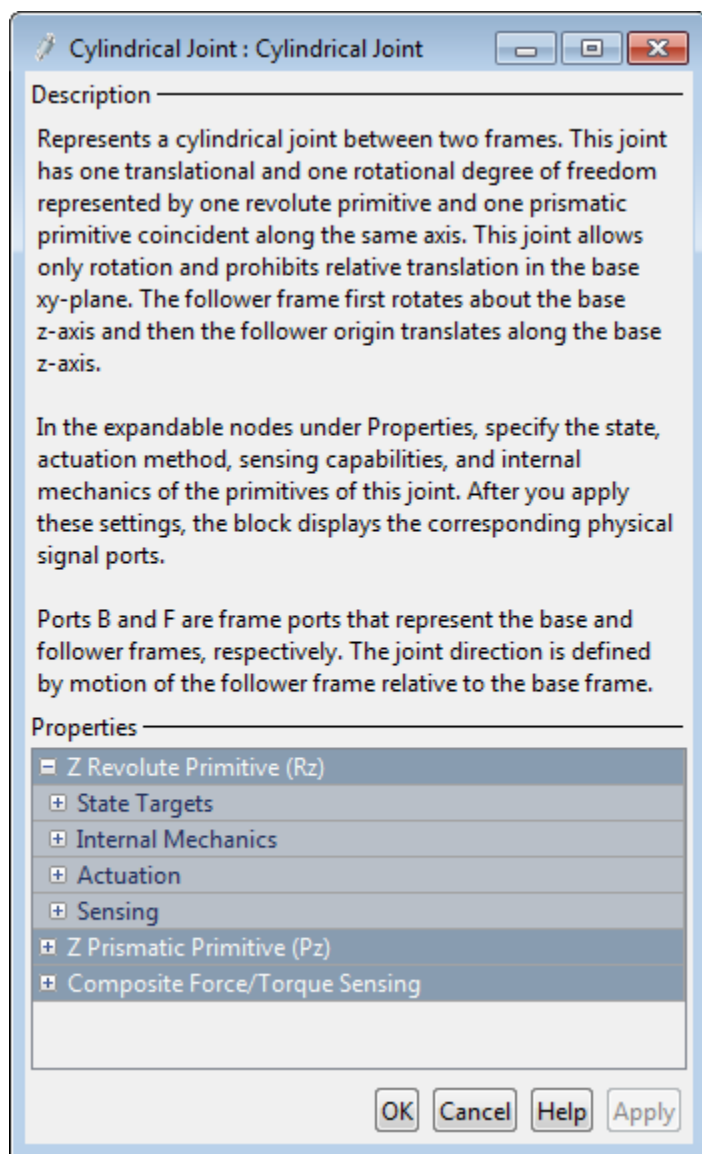
Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types: Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.



## Revolute Primitive: State Targets

Specify the revolute primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative rotation angle, measured about the joint primitive axis, of the follower frame with respect to the base frame. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative angular velocity, measured about the joint primitive axis, of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is **deg** for position and **deg/s** for velocity.

## Revolute Primitive: Internal Mechanics

Specify the revolute primitive internal mechanics. Internal mechanics include linear spring torques, accounting for energy storage, and linear damping torques, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

### Equilibrium Position

Enter the spring equilibrium position. This is the rotation angle between base and follower frames at which the spring torque is zero. The default value is 0. Select or enter a physical unit. The default is **deg**.

### Spring Stiffness

Enter the linear spring constant. This is the torque required to rotate the joint primitive by a unit angle. The default is 0. Select or enter a physical unit. The default is  $\text{N}\cdot\text{m}/\text{deg}$ .

### Damping Coefficient

Enter the linear damping coefficient. This is the torque required to maintain a constant joint primitive angular velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is  $\text{N}\cdot\text{m}/(\text{deg}/\text{s})$ .

## Revolute Primitive: Actuation

Specify actuation options for the revolute joint primitive. Actuation modes include **Torque** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Input signals are resolved in the base frame.

### Torque

Select an actuation torque setting. The default setting is **None**.

Actuation Torque Setting	Description
None	No actuation torque.
Provided by Input	Actuation torque from physical signal input. The signal provides the torque acting on the follower frame with respect to the base frame about the joint primitive axis. An equal and opposite torque acts on the base frame.

Actuation Torque Setting	Description
Automatically computed	Actuation torque from automatic calculation. SimMechanics computes and applies the actuation torque based on model dynamics.

### Motion

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

### Revolute Primitive: Sensing

Select the variables to sense in the revolute joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

#### Position

Select this option to sense the relative rotation angle of the follower frame with respect to the base frame about the joint primitive axis.

#### Velocity

Select this option to sense the relative angular velocity of the follower frame with respect to the base frame about the joint primitive axis.

#### Acceleration

Select this option to sense the relative angular acceleration of the follower frame with respect to the base frame about the joint primitive axis.

**Actuator Torque**

Select this option to sense the actuation torque acting on the follower frame with respect to the base frame about the joint primitive axis.

**Prismatic Primitive: State Targets**

Specify the prismatic primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

**Specify Position Target**

Select this option to specify the desired joint primitive position at time zero. This is the relative position, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

**Specify Velocity Target**

Select this option to specify the desired joint primitive velocity at time zero. This is the relative velocity, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. It is resolved in the base frame. Selecting this option exposes priority and value fields.

**Priority**

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---



**Value**

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is m for position and m/s for velocity.

**Prismatic Primitive: Internal Mechanics**

Specify the prismatic primitive internal mechanics. Internal mechanics include linear spring forces, accounting for energy storage, and damping forces, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

**Equilibrium Position**

Enter the spring equilibrium position. This is the distance between base and follower frame origins at which the spring force is zero. The default value is 0. Select or enter a physical unit. The default is m.

**Spring Stiffness**

Enter the linear spring constant. This is the force required to displace the joint primitive by a unit distance. The default is 0. Select or enter a physical unit. The default is N/m.

**Damping Coefficient**

Enter the linear damping coefficient. This is the force required to maintain a constant joint primitive velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is N/(m/s).

**Prismatic Primitive: Actuation**

Specify actuation options for the prismatic joint primitive. Actuation modes include **Force** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Actuation signals are resolved in the base frame.

**Force**

Select an actuation force setting. The default setting is **None**.

Actuation Force Setting	Description
None	No actuation force.

Actuation Force Setting	Description
Provided by Input	Actuation force from physical signal input. The signal provides the force acting on the follower frame with respect to the base frame along the joint primitive axis. An equal and opposite force acts on the base frame.
Automatically computed	Actuation force from automatic calculation. SimMechanics computes and applies the actuation force based on model dynamics.

**Motion**

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

**Prismatic Primitive: Sensing**

Select the variables to sense in the prismatic joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

**Position**

Select this option to sense the relative position of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Velocity**

Select this option to sense the relative velocity of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Acceleration**

Select this option to sense the relative acceleration of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Actuator Force**

Select this option to sense the actuation force acting on the follower frame with respect to the base frame along the joint primitive axis.

**Composite Force/Torque Sensing**

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

**Direction**

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

**Resolution Frame**

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

**Constraint Force**

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector  $[fc_x, fc_y, fc_z]$  acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

**Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector  $[tc_x, tc_y, tc_z]$  acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

**Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector  $[ft_x, ft_y, ft_z]$ . Selecting this option causes the block to expose physical signal port *ft*.

**Total Torque**

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.

- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port  $tt$ .

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
f	Force	Actuation input	Scalar
t	Torque	Actuation input	Scalar
p	Position	Sensing output	Scalar
v	Velocity	Sensing output	Scalar
a	Acceleration	Sensing output	Scalar
q	Rotation angle	Sensing output	Scalar
w	Angular velocity	Sensing output	Scalar
b	Angular acceleration	Sensing output	Scalar

## See Also

Prismatic Joint | Revolute Joint

## **More About**

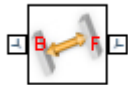
- “Actuating and Sensing Using Physical Signals”
- “Motion Sensing”
- “Rotational Measurements”
- “Translational Measurements”

# Distance Constraint

Fixed distance between two frame origins

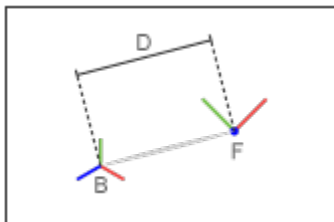
## Library

Constraints



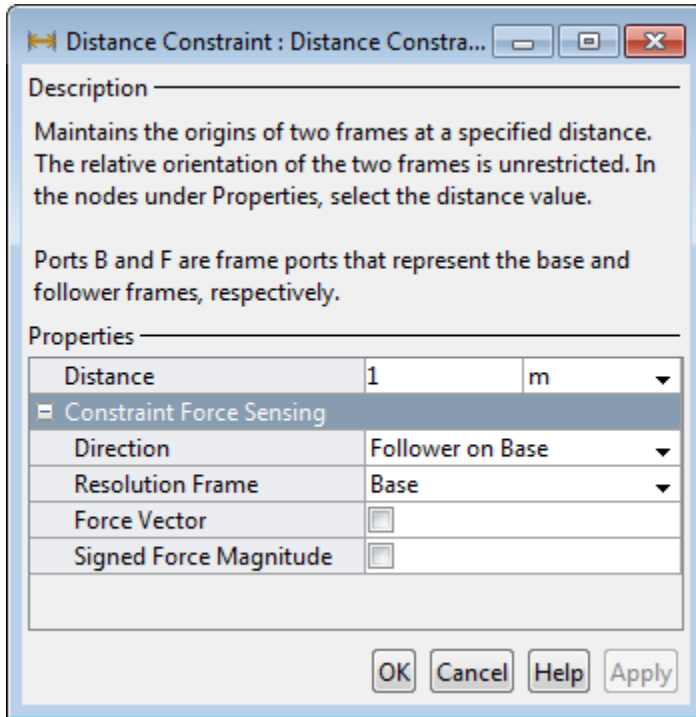
## Description

This block applies a fixed distance between the origins of the base and follower port frames. The frames lose one translational degree of freedom with respect to each other. The constraint distance between the frame origins, labeled  $D$  in the figure, must be greater than zero.



The block provides constraint force sensing in the form of a vector or a signed magnitude. These quantities are contained in physical signals that the block outputs through Simscape™ PS ports. The constraint force is the force required to maintain the specified distance between the port frame origins.

## Dialog Box and Parameters



### Distance

Constraint distance between the base and follower frame origins. The distance must be greater than zero. For a distance of zero, use a Spherical Joint or Gimbal Joint block instead. The default value is 1 m.

### Constraint Force Sensing

Select whether to compute and output the distance constraint force vector and its signed magnitude. The distance constraint force is the force that the block must apply in order to maintain the distance you specify between the base and follower port frames.

### Direction

Constraint forces act in pairs. As expressed by Newton's third law of motion, if the base port frame exerts a constraint force on the follower port frame, then the follower



port frame must exert an equal and opposite force on the base port frame. Select which of the two constraint forces to sense:

- **Follower on Base** — Sense the constraint force that the follower port frame exerts on the base port frame.
- **Base on Follower** — Sense the constraint force that the base port frame exerts on the follower port frame.

### **Resolution Frame**

The block expresses the constraint force vector in terms of its Cartesian vector components. The splitting of a vector into vector components is known as vector resolution. The frame whose axes define the vector component directions is known as the resolution frame. Select whether to resolve the constraint force vector in the base or follower port frame.

### **Force Vector**

Compute and output the Cartesian components of the distance constraint force vector. The output signal is a three-dimensional vector,  $[f_x, f_y, f_z]$ .

### **Signed Force Magnitude**

Compute and output the magnitude of the distance constraint force, including its sign.

## **Ports**

The block provides two frame ports:

- **B** — Base frame port
- **F** — Follower frame port

In addition, the block provides two physical signal output ports:

- **f** — Distance constraint force vector
- **fm** — Signed magnitude of the distance constraint force

## **See Also**

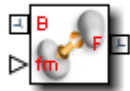
Angle Constraint

## Internal Force

General force acting reciprocally between two frame origins

### Library

Forces and Torques



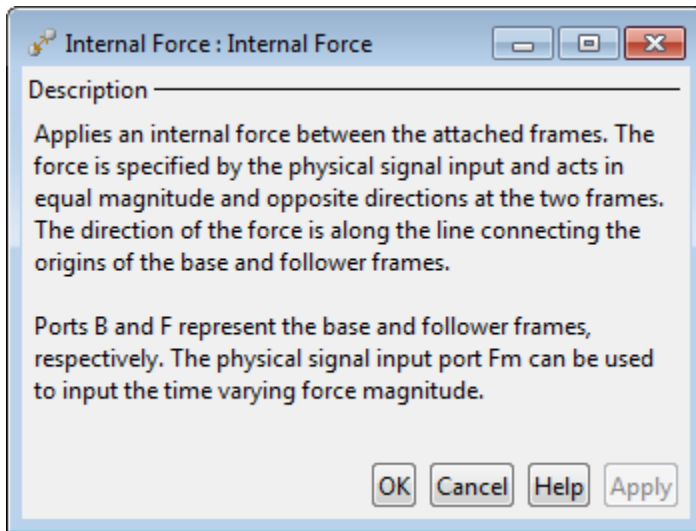
### Description

This block represents a general force pair acting reciprocally between base and follower frame origins. The two forces in the pair have equal magnitude but opposite directions. One force acts on the base frame origin, along the vector connecting follower to base frame origins. The other force acts on the follower frame origin, along the vector connecting base to follower frame origins.

To specify the internal force, the block provides physical signal port **fm**. A positive input signal represents a repulsive force, which pushes base and follower frame origins apart. A negative input signal represents an attractive force, which pulls base and follower frame origins together. If at any time the two frame origins are coincident, the internal force direction becomes undefined and SimMechanics might stop with an error.

### Dialog Box and Parameters

The dialog box contains no parameters that you can specify. To specify the internal force, connect a physical signal to port **fm**.



## Ports

This block contains frame ports **B** and **F**, representing base and follower port frames, respectively. A physical signal port, **fm**, provides the means to specify the internal force acting between the two port frames.

## See Also

External Force and Torque | Spring and Damper Force | Inverse Square Law Force

## More About

- “Actuating and Sensing Using Physical Signals”

# External Force and Torque

General force and torque arising outside the modeled system

## Library

Forces and Torques



## Description

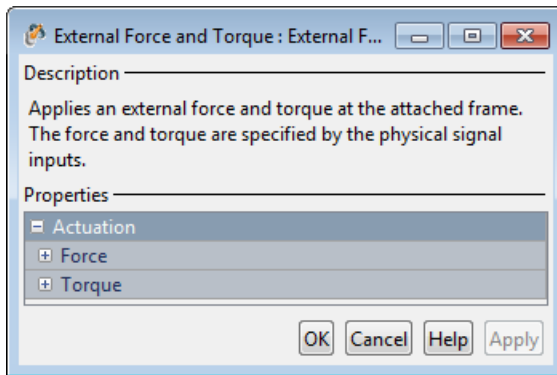
This block represents a general force and torque that an external agency applies on a rigid body frame. The force and torque input can be constant or it can vary with time. The block provides a set of physical signal ports that you use to specify this input. The ports are hidden by default. Selecting an actuation mode exposes the corresponding physical signal port.

Each force and torque input acts on the origin of the follower frame in a direction that depends on the chosen force type and resolution frame. A force/torque vector component acts along/about the associated Cartesian axis, resolved in the chosen resolution frame. For example, the **Force(X)** input acts along the X axis of the resolution frame. A complete force/torque vector acts in the general direction that its components specify, resolved in the chosen resolution frame.

A force input with a positive value acts along the positive direction of the associated axis. A torque input with a positive value acts about the associated axis according to the right-hand rule.

## Dialog Box and Parameters

The dialog box contains a **Properties** area with expandable **Force** and **Torque** Sections. Expanding these sections exposes the force and torque types that you can select.



## Actuation: Force

Select the force inputs to specify and the frame to resolve them in.

### Force Resolution Frame

Select the frame to resolve each force signal in. The components of the force vector align with the axes of the resolution frame. The default setting is **Attached Frame**. The table summarizes the resolution frames that you can select.

Resolution Frame	Description
World	Resolve each force component in the World frame of the model.
Attached Frame	Resolve each force component in the follower frame of the External Force and Torque block. The follower frame is the attached frame of the block.

### Force Inputs

Select the force inputs to specify. Options include the complete force vector and the separate components of that vector. Selecting a force input exposes the physical signal port associated with that input. Use that port to specify the force input via physical signals. The table summarizes the force inputs that you can select.

Force Input	Description	Input
<b>Force(X)</b> , <b>Force(Y)</b> , <b>Force(Z)</b>	Specify separately the force components acting on	Scalar

Force Input	Description	Input
	the origin of the follower frame along the X, Y, and Z axes of the resolution frame	
<b>Force</b>	Specify the complete force vector [fx fy fz] acting on the origin of the follower frame along the X, Y, and Z axes of the resolution frame	Three-element vector

## Actuation: Torque

Select the torque inputs to specify and the frame to resolve them in.

### Torque Resolution Frame

Select the frame to resolve each torque signal in. The components of the torque vector align with the axes of the resolution frame. The default setting is **Attached Frame**. The table summarizes the resolution frames that you can select.

Resolution Frame	Description
World	Resolve each torque component in the World frame of the model.
Attached Frame	Resolve each torque component in the follower frame of the External Force and Torque block. The follower frame is the attached frame of the block.

### Torque Inputs

Select the torque inputs to specify. Options include the complete torque vector and the separate components of that vector. Selecting a torque input exposes the physical signal port associated with that input. Use that port to specify the torque input via physical signals. The table summarizes the torque inputs that you can select.

Force Input	Description	Input
<b>Torque(X), Torque(Y), Torque(Z)</b>	Specify separately the torque components acting	Scalar

Force Input	Description	Input
	on the origin of the follower frame about the X, Y, and Z axes of the resolution frame	
<b>Torque</b>	Specify the complete torque vector [fx fy fz] acting on the origin of the follower frame about a general direction in the resolution frame	Three-element vector

## Ports

The block contains frame port F, representing the follower frame. Selecting an actuation mode exposes additional physical signal ports. Use the ports to input the selected actuation signals.

Each physical signal port has a unique label. The table identifies the actuation modes that the port labels correspond to.

Port Label	Description
fx, fy, fz	Force vector components acting on the origin of the follower frame along the X, Y, and Z axes, respectively
f	Force vector [fx, fy, fz] acting on the origin of the follower frame along a general direction [X Y Z]
tx, ty, tz	Torque vector components acting on the origin of the follower frame about the X, Y, and Z axes, respectively
t	Torque vector [tx ty tz] acting on the origin of the follower frame about a general direction [X Y Z]

## See Also

Inverse Square Law Force | Spring and Damper Force | Internal Force

## **More About**

- “Actuating and Sensing Using Physical Signals”



# Gimbal Joint

Joint with three revolute primitives

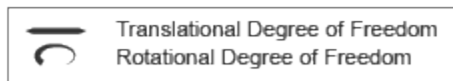
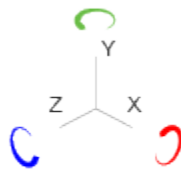
## Library

Joints



## Description

This block represents a joint with three rotational degrees of freedom. Three revolute primitives provide the three rotational degrees of freedom. The base and follower frame origins remain coincident during simulation.



## Joint Degrees of Freedom

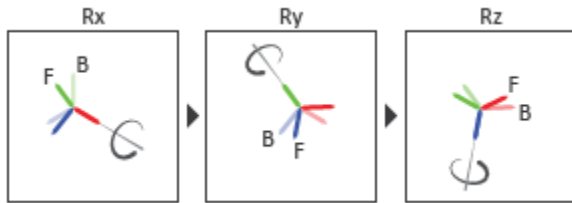
The joint block represents motion between the base and follower frames as a sequence of time-varying transformations. Each joint primitive applies one transformation in this sequence. The transformation translates or rotates the follower frame with respect to the joint primitive base frame. For all but the first joint primitive, the base frame coincides with the follower frame of the previous joint primitive in the sequence.

At each time step during the simulation, the joint block applies the sequence of time-varying frame transformations in this order:

1 Rotation:

- a About the X axis of the X Revolute Primitive (Rx) base frame.
- b About the Y axis of the Y Revolute Primitive (Ry) base frame. This frame is coincident with the X Revolute Primitive (Rx) follower frame.
- c About the Z axis of the Z Revolute Primitive (Rz) base frame. This frame is coincident with the Y Revolute Primitive (Ry) follower frame.

The figure shows the sequence in which the joint transformations occur at a given simulation time step. The resulting frame of each transformation serves as the base frame for the following transformation. Because 3-D rotation occurs as a sequence, it is possible for two axes to align, causing to the loss of one rotational degree of freedom. This phenomenon is known as gimbal lock.



**Joint Transformation Sequence**

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

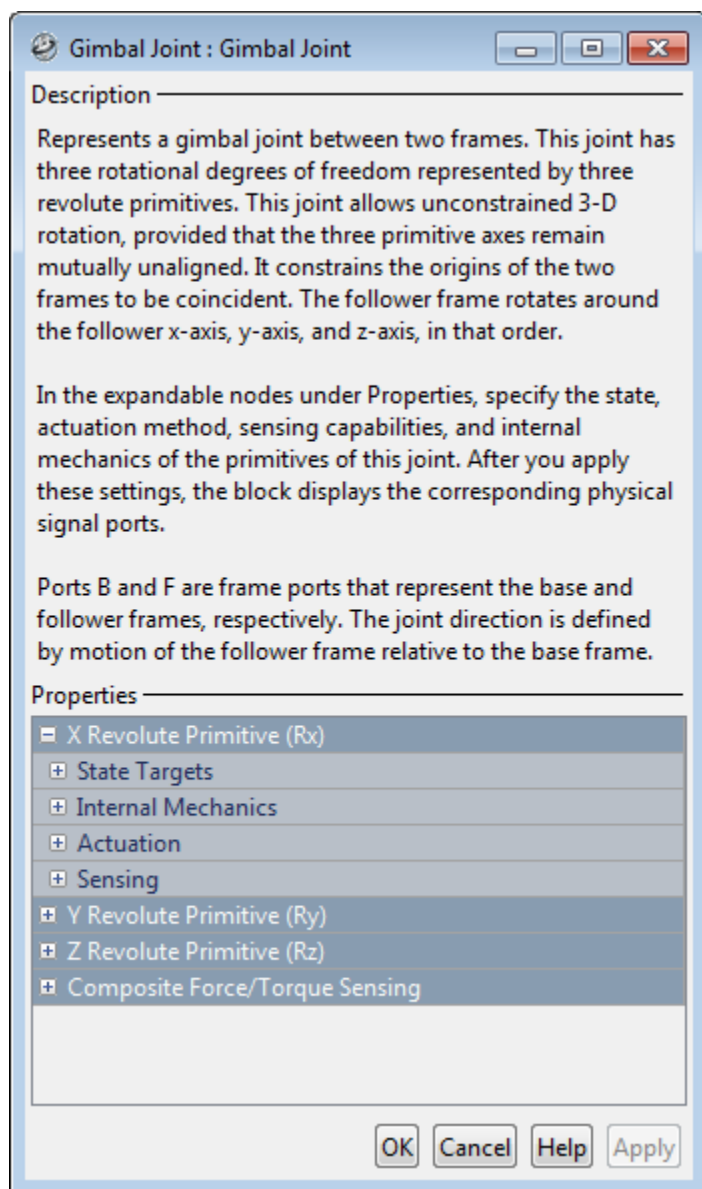
Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types: Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.



## Revolute Primitive: State Targets

Specify the revolute primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative rotation angle, measured about the joint primitive axis, of the follower frame with respect to the base frame. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative angular velocity, measured about the joint primitive axis, of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is **deg** for position and **deg/s** for velocity.

## Revolute Primitive: Internal Mechanics

Specify the revolute primitive internal mechanics. Internal mechanics include linear spring torques, accounting for energy storage, and linear damping torques, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

### Equilibrium Position

Enter the spring equilibrium position. This is the rotation angle between base and follower frames at which the spring torque is zero. The default value is 0. Select or enter a physical unit. The default is **deg**.

### Spring Stiffness

Enter the linear spring constant. This is the torque required to rotate the joint primitive by a unit angle. The default is 0. Select or enter a physical unit. The default is **N\*m/deg**.

### Damping Coefficient

Enter the linear damping coefficient. This is the torque required to maintain a constant joint primitive angular velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is **N\*m/(deg/s)**.

## Revolute Primitive: Actuation

Specify actuation options for the revolute joint primitive. Actuation modes include **Torque** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Input signals are resolved in the base frame.

### Torque

Select an actuation torque setting. The default setting is **None**.

Actuation Torque Setting	Description
None	No actuation torque.
Provided by Input	Actuation torque from physical signal input. The signal provides the torque acting on the follower frame with respect to the base frame about the joint primitive axis. An equal and opposite torque acts on the base frame.

Actuation Torque Setting	Description
Automatically computed	Actuation torque from automatic calculation. SimMechanics computes and applies the actuation torque based on model dynamics.

### Motion

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

### Revolute Primitive: Sensing

Select the variables to sense in the revolute joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

#### Position

Select this option to sense the relative rotation angle of the follower frame with respect to the base frame about the joint primitive axis.

#### Velocity

Select this option to sense the relative angular velocity of the follower frame with respect to the base frame about the joint primitive axis.

#### Acceleration

Select this option to sense the relative angular acceleration of the follower frame with respect to the base frame about the joint primitive axis.

### **Actuator Torque**

Select this option to sense the actuation torque acting on the follower frame with respect to the base frame about the joint primitive axis.

## **Composite Force/Torque Sensing**

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

### **Direction**

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

### **Resolution Frame**

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

### **Constraint Force**

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint



force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector  $[f_{c_x}, f_{c_y}, f_{c_z}]$  acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

### **Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector  $[t_{c_x}, t_{c_y}, t_{c_z}]$  acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

### **Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector  $[f_{t_x}, f_{t_y}, f_{t_z}]$ . Selecting this option causes the block to expose physical signal port *ft*.

### **Total Torque**

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.
- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector [ $tt_x$ ,  $tt_y$ ,  $tt_z$ ]. Selecting this option causes the block to expose physical signal port  $tt$ .

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
t	Torque	Actuation input	Scalar
q	Rotation angle	Sensing output	Scalar
w	Angular velocity	Sensing output	Scalar
b	Angular acceleration	Sensing output	Scalar

## See Also

Revolute Joint | Spherical Joint | Bushing Joint

## More About

- “Actuating and Sensing Using Physical Signals”
- “Motion Sensing”
- “Rotational Measurements”

# Graphic

Visual marker with graphic properties

## Library

Body Elements

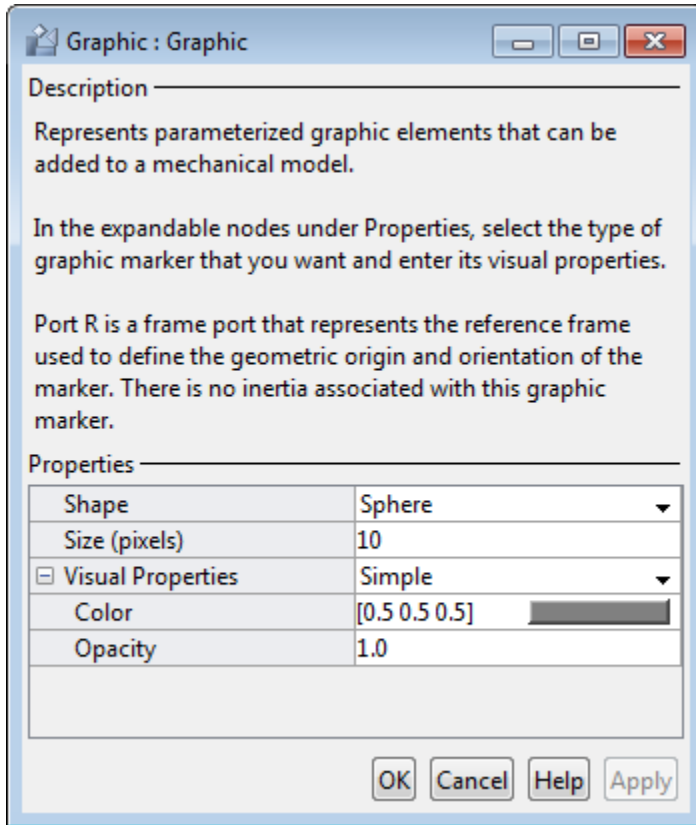


## Description

This block represents a simple 3-D marker with shape, color, and opacity. Shapes include sphere, cube, and frame. A color palette provides a selection of predefined colors. The marker has no inertial properties and therefore does not affect model dynamics. The figure shows the different markers that you can select.



## Dialog Box and Parameters



### Shape

Geometric shape of the graphic marker. Options include Cube, Frame, and Sphere. The default setting is Sphere.

### Size

Absolute size of the graphic marker in pixels. Changing the zoom level in the model visualization pane has no effect on the apparent marker size. The default value is 10.

### Visual Properties

Color specification type. Options include **Simple** and **Advanced**. Select **Simple** to specify the solid color and opacity. Select **Advanced** to add lighting effects such as specular reflections and light emission.

**Simple: Color**

[R G B] color vector. This vector contains the red (R), green (G), and blue (B) contents of the solid color on a scale of 0–1. The default vector is [0.5 0.5 0.5]. A color picker provides an alternative means of defining the solid color.

**Simple: Opacity**

Degree to which the solid obscures model components positioned behind it. The opacity value can range from 0 to 1. An opacity of 0 makes the solid completely translucent, while an opacity of 1 makes the solid completely opaque. The default value is 1.0.

**Advanced: Diffuse Color**

[R G B] or [R G B A] diffuse color vector. The diffuse color is the apparent color of the solid under direct white light. The color vector contains the red (R), green (G), and blue (B) contents of the diffuse color on a scale of 0–1. It can include an optional opacity value (A), also on a scale of 0–1. The default vector is [0.5 0.5 0.5].

**Advanced: Specular Color**

[R G B] or [R G B A] specular color vector. The specular color is the color of the glossy highlights on the solid surface. The color vector contains the red (R), green (G), and blue (B) contents of the specular color on a scale of 0–1. It can include an optional opacity value (A), also on a scale of 0–1. The default vector is [0.5 0.5 0.5 1.0].

**Advanced: Ambient Color**

[R G B] or [R G B A] ambient color vector. The ambient color is the apparent color of the solid under indirect ambient light. The color vector contains the red (R), green (G), and blue (B) contents of the ambient color on a scale of 0–1. It can include an optional opacity value (A), also on a scale of 0–1. The default vector is [0.15 0.15 0.15 1.0].

**Advanced: Emissive Color**

[R G B] or [R G B A] emissive color vector. The emissive color is the color of light generated by the solid. The color vector contains the red (R), green (G), and blue (B) contents of the emissive color on a scale of 0–1. It can include an optional opacity value (A), also on a scale of 0–1. The default vector is [0.0 0.0 0.0 1.0].

**Advanced: Shininess**

Sharpness of the specular highlights on the solid surface. The shininess value can range from 0 to 128. A low shininess value produces large specular highlights

with a gradual falloff in intensity. A large shininess value produces small specular highlights with a sharp falloff in intensity. The default value is 75.

## **Ports**

This block contains frame port R, representing the graphic reference frame.

## **See Also**

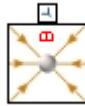
Inertia | Solid

# Gravitational Field

Field of force due to point mass

## Library

Forces and Torques



## Description

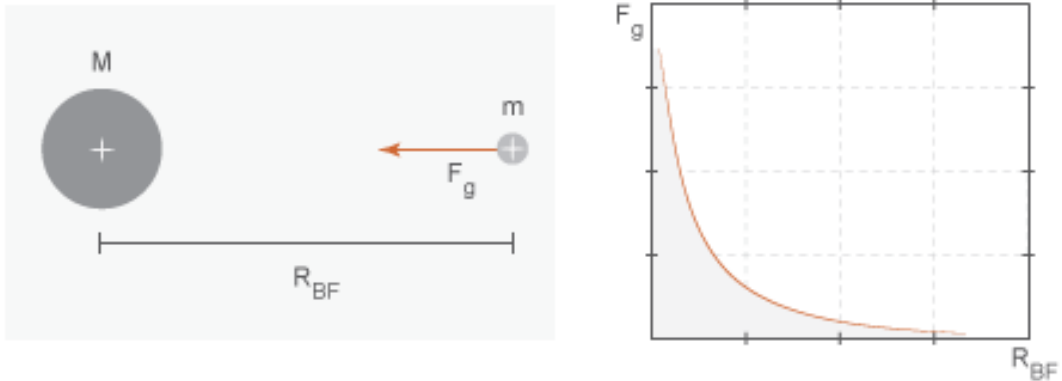
This block represents the gravitational field of a point mass. This field applies a gravitational force at the center of mass of each rigid body. The force magnitude decays with the square distance from the field origin, coincident with the base port frame origin. The force on a rigid body follows from Newton's universal gravitation law:

$$F_g = -G \frac{Mm}{R_{BF}^2},$$

where:

- $F_g$  is the force that the gravitational field exerts on a given rigid body.
- $G$  is the universal gravitational constant,  $6.67384 \times 10^{-11} \text{ m}^3\text{kg}^{-1}\text{s}^{-2}$ .
- $M$  is the total mass generating the gravitational field.
- $m$  is the total mass of the rigid body the gravitational force acts upon.
- $R_{BF}$  is the distance between the source mass position and the rigid body center of mass.

The figure shows these variables. The plot shows the inverse square dependence between the gravitational force and distance.



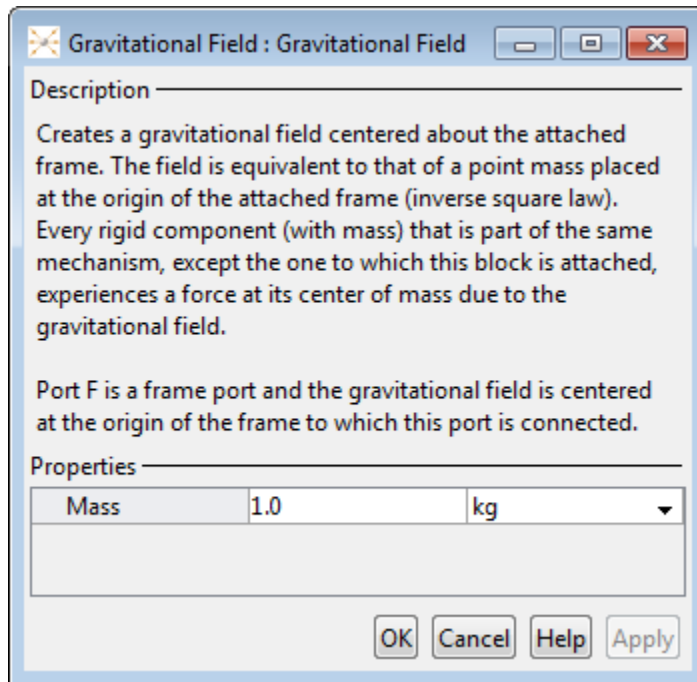
The source mass can be positive or negative. Combine multiple instances of this block to model the gravitational effects that positive and negative mass disturbances impose on a stronger gravitational field, such as a reduction in the gravitational pull of a planet due to a concentration of low-density material along a portion of its surface.

This block excludes the gravitational forces that other rigid bodies exert on the field source mass. To include these forces, you can connect Gravitational Field blocks to other rigid bodies in the model. Alternatively, you can use the Inverse Square Law Force block to model the gravitational forces between a single pair of rigid bodies.

The gravitational field is time invariant. To specify a time-varying, spatially uniform field, use the Mechanism Configuration block.



## Dialog Box and Parameters



### Mass

Total mass generating the gravitational field. The resulting gravitational forces are directly proportional to this mass. This mass adds no inertia to the model. The default value for the mass parameter is 1.0 kg.

### Ports

Frame port B represents a frame with origin at the point mass responsible for the gravitational field.

### See Also

Inverse Square Law Force | Mechanism Configuration

### **Related Examples**

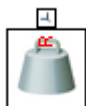
- “Model Planetary Orbit Due to Gravity”

# Inertia

Inertial properties of a solid or mass disturbance

## Library

Body Elements



## Description

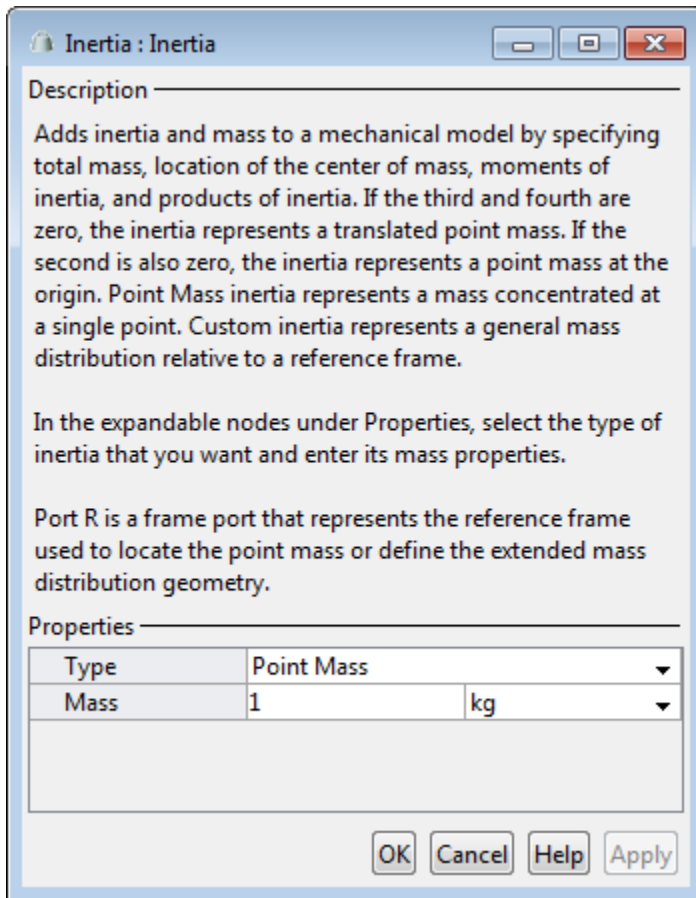
This block represents the inertial properties of a solid. The solid can be a point mass or a 3-D mass distribution. To represent the inertial properties of a point mass, the dialog box provides a **Point Mass** parameterization. Inertial parameters of a point mass include only the total mass of the solid. To represent the inertial properties of a 3-D mass distribution, the dialog box provides a **Custom** parameterization. Inertial parameters of a custom inertia include the total mass of the solid, its center of mass, as well as its moments and products of inertia.

This block can also represent a mass disturbance in a model. The disturbance can have positive or negative inertia. A disturbance with negative inertia reduces the total inertia of the rigid body the block connects to. A disturbance with positive inertia increases the total inertia of the rigid body the block connects to. Use this block to adjust the total inertia of a rigid body.

The visualization pane of Mechanics Explorer identifies the position of an Inertia element with the inertia icon .

## Dialog Box and Parameters

The dialog box contains a **Properties** area with inertia options and parameters.



### Type

Select a method to specify inertia. The default is Point Mass.

Type	Description
Point Mass	Treat the inertia as a mass with zero volume. The point mass is located at the reference frame origin.
Custom	Manually specify all inertial parameters, including mass, center of mass, and moments and products of inertia.

**Point Mass/Custom: Mass**

Enter the total mass of the solid. Select a physical unit. The default is 1 Kg.

**Custom: Center of Mass**

Enter the center of mass coordinates with respect to the solid reference frame in the order [X Y Z]. In a uniform gravitational field, the center of mass coincides with the center of gravity. Select a physical unit. The default is [0 0 0].

**Custom: Moments of Inertia**

Enter the mass moments of inertia of the solid element in the order [ $I_{xx}$ ,  $I_{yy}$ ,  $I_{zz}$ ]. Each moment of inertia must refer to a frame whose axes are parallel to the block reference frame axes and whose origin is coincident with the solid center of mass. The moments of inertia are the diagonal elements of the solid inertia tensor,

$$\begin{pmatrix} I_{xx} & & \\ & I_{yy} & \\ & & I_{zz} \end{pmatrix},$$

where:

- $$I_{xx} = \int_V (y^2 + z^2) dm$$
- $$I_{yy} = \int_V (x^2 + z^2) dm$$
- $$I_{zz} = \int_V (x^2 + y^2) dm$$

Select a physical unit. The default is [1 1 1] kg\*m<sup>2</sup>.

**Custom: Products of Inertia**

Enter the mass products of inertia of the solid element in the order [ $I_{yz}$ ,  $I_{zx}$ ,  $I_{xy}$ ]. Each product of inertia must refer to a frame whose axes are parallel to the block reference frame axes and whose origin is coincident with the solid center of mass. The products of inertia are the off-diagonal elements of the solid inertia tensor,

$$\begin{pmatrix} & I_{xy} & I_{zx} \\ I_{xy} & & I_{yz} \\ I_{zx} & I_{yz} & \end{pmatrix},$$

where:

- $I_{yz} = -\int_V yz \, dm$
- $I_{zx} = -\int_V zx \, dm$
- $I_{xy} = -\int_V xy \, dm$

Select a physical unit. The default is [0 0 0] kg\*m^2.

## Ports

This block contains frame port R, representing the inertia reference frame.

## See Also

Graphic | Solid

## More About

- “Solid Inertia”

# Inverse Square Law Force

Force proportional to the inverse square distance between two frame origins

## Library

Forces and Torques



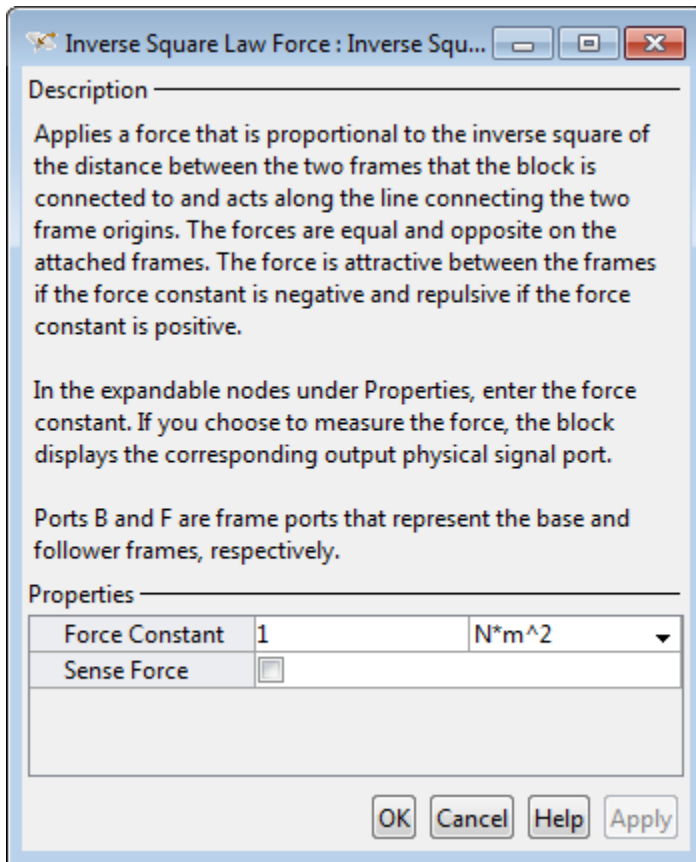
## Description

This block represents a force pair that is inversely proportional to the square distance between the base and follower frame origins. The two forces in the pair have equal magnitude but opposite directions. One force acts on the base frame origin, along the vector connecting the follower to base frame origins. The other force acts on the follower frame origin, along the vector connecting base to follower frame origins.

The value of the force depends on a force constant that you specify. A positive force constant represents a repulsive force that pushes the two frame origins apart. A negative force constant represents an attractive force that pulls the two frame origins together.

## Dialog Box and Parameters

The dialog box contains a **Properties** area with one parameter that you can specify and one option that you can select.



### Force Constant

Specify the proportionality constant of the inverse square law force. This constant is a lumped parameter that encodes the dependence of the force magnitude on the inverse square distance between the two frame origins. The default value is 1. Select or specify a physical unit.

### Sense Force

Select the check box to sense the signed magnitude of the inverse square law force acting between the two frame origins. The block exposes an additional physical signal port to output the force signal. The output signal is a scalar value. This value is positive if the force is repulsive; it is negative if the force is attractive.



## Ports

The block contains frame ports B and F, representing base and follower frames, respectively.

Selecting **Sense Force** in the block dialog box exposes an additional physical signal port, **fm**.

## See Also

External Force and Torque | Internal Force | Spring and Damper Force

## More About

- “Actuating and Sensing Using Physical Signals”

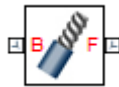
# Lead Screw Joint

Joint with coupled rotational and translational degrees of freedom

## Library

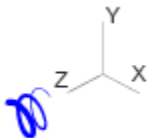
Joints

## Description



This block represents a joint with one rotational degree of freedom coupled with one translational degree of freedom. The coupling between the two degrees of freedom ensures that whenever the joint frames rotate relative to each other, they also translate by a commensurate amount and vice-versa. The joint lead determines the translation distance associated with a unit rotation angle while the joint direction determines whether a positive angle results in a positive or negative translation.

During assembly and simulation, the joint aligns the Z axes of its port frames. The common Z axis functions as the rotation and translation axis. Whenever the joint frames rotate, they do so about the common Z axis, and whenever the joint frames translate, they do so along the common Z axis. You can orient the motion axis in a different direction by applying rotation transforms to the joint frames through Rigid Transform blocks.



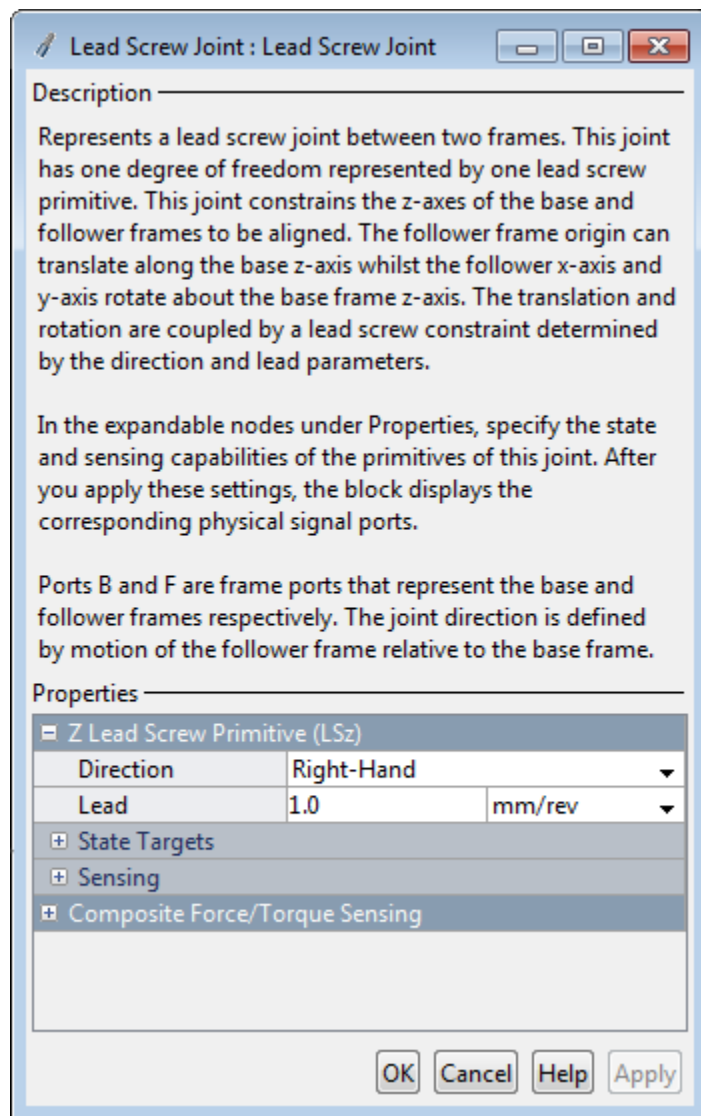
## Joint Degrees of Freedom

A set of optional state targets guide assembly for the joint primitive. Targets include position and velocity. You can specify these based on the relative rotation or translation

between the joint frames. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

Each joint primitive has a set of optional sensing ports. These ports provide physical signal outputs that measure joint primitive motion. Variables that you can sense include those describing translational motion, rotational motion, and constraint forces and torques.

## Dialog Box and Parameters



## Lead Screw Primitive

### Direction

Handedness of motion between the joint frames. Motion is right-handed if a positive rotation leads to a positive translation and left-handed if a positive rotation leads to a negative translation. The default setting is **Right - Hand**.

### Lead

Translation distance between the joint frames due to a unit rotation angle. The larger the lead, the longer the frames must translate before completing a full revolution. The default value is **1.0 mm/rev**.

## Lead Screw Primitive: State Targets

Specify the lead screw primitive state targets and their priority levels. A state target is the desired value for one of the joint state variables—position or velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be satisfied.

### Specify Position Target

Desired joint primitive position at the start of simulation. This is the relative position, rotational or translational, of the follower frame relative to the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Desired joint velocity at the start of simulation. This is the relative velocity, rotational or translational, of the follower frame relative to the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

**Based On**

Motion type that the state target is based on. Options include **Rotation** and **Translation**. The default setting is **Translation**.

**Value**

Desired value of the position or velocity state target. The default value is 0.

**Lead Screw Primitive: Sensing**

Select the variables to sense in the lead screw primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame.

Variable	Description
Rotation: <b>Position</b>	Rotation angle of the follower frame relative to the base frame about the common Z axis. Selecting the check box exposes a physical signal port labeled q.
Rotation: <b>Velocity</b>	Rotational velocity of the follower frame relative to the base frame about the common Z axis. Selecting the check box exposes a physical signal port labeled w.
Rotation: <b>Acceleration</b>	Rotational acceleration of the follower frame relative to the base frame about the common Z axis. Selecting the check box exposes a physical signal port labeled b.
Translation: <b>Position</b>	Offset distance of the follower frame relative to the base frame along the common Z axis. Selecting the check box exposes a physical signal port labeled p.
Translation: <b>Velocity</b>	Translational velocity of the follower frame relative to the base frame along the

Variable	Description
	common Z axis. Selecting the check box exposes a physical signal port labeled v.
Translation: <b>Acceleration</b>	Translational acceleration of the follower frame relative to the base frame along the common Z axis. Selecting the check box exposes a physical signal port labeled a.

## Composite Force/Torque Sensing

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

### Direction

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

### Resolution Frame

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

### Constraint Force

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector [ $f_{c_x}$ ,  $f_{c_y}$ ,  $f_{c_z}$ ] acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

### **Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector [ $t_{c_x}$ ,  $t_{c_y}$ ,  $t_{c_z}$ ] acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

### **Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector [ $f_{t_x}$ ,  $f_{t_y}$ ,  $f_{t_z}$ ]. Selecting this option causes the block to expose physical signal port *ft*.

### **Total Torque**

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.
- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.



- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port  $tt$ .

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description
q	Rotation angle
w	Angular velocity
b	Angular acceleration
p	Position
v	Linear velocity
a	Linear acceleration
fc	Constraint force
tc	Constraint torque
ft	Total force
tt	Total torque

## See Also

Prismatic Joint | Revolute Joint

# Mechanism Configuration

Mechanism-wide simulation and mechanical parameters

## Library

Utilities

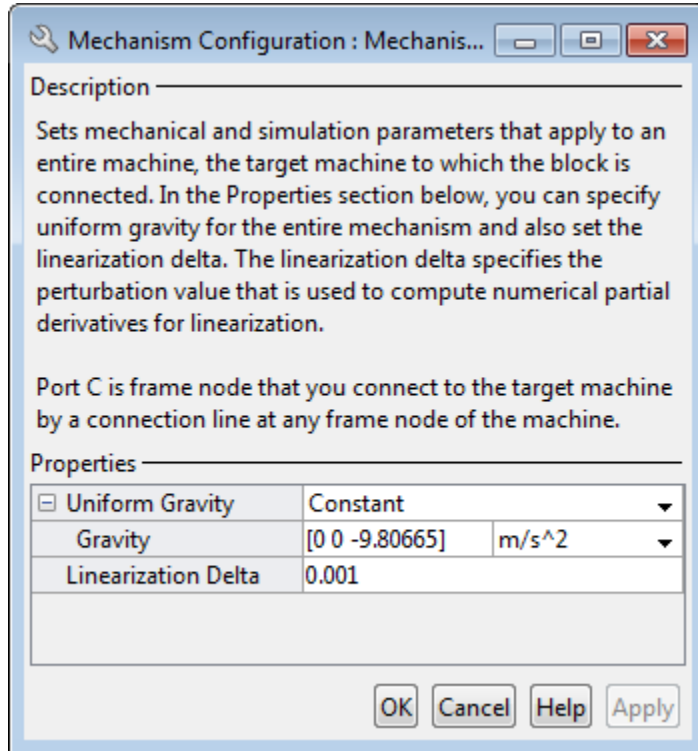


## Description

This block provides mechanical and simulation parameters to a mechanism, i.e., a self-contained group of interconnected SimMechanics blocks. Parameters include gravity and a linearization delta for computing numerical partial derivatives during linearization. These parameters apply only to the target mechanism, i.e., the mechanism that the block connects to.

The Mechanism Configuration block is optional. If you omit it, the gravitational acceleration vector is set to zero. Use only one instance of this block per mechanism, setting uniform gravity to **None** if that mechanism contains one or more Gravitational Field blocks.

## Dialog Box and Parameters



### Uniform Gravity

Type of gravitational acceleration vector in effect at the target mechanism. Options include:

- **None** — Specify zero gravity. The block automatically applies the gravitational acceleration vector [0 0 0] to the target mechanism. If the mechanism contains one or more Gravitational Field blocks, you must select this option.
- **Constant** — Specify a gravitational acceleration vector that remains constant in space and in time. Selecting this option exposes an additional parameter, **Gravity**. If the target mechanism contains one or more Gravitational Field blocks, you must select **None** instead.
- **Time-Varying** — Specify a gravitational acceleration vector that remains constant in space but varies in time. Selecting this option exposes a physical

signal port. Use that port to specify the time-varying gravitational acceleration vector. If the target mechanism contains one or more Gravitational Field blocks, you must select **None** instead.

**Gravity**

Nominal acceleration vector due to gravity. The block resolves this vector in the mechanism World frame. The default vector is  $[0 \ 0 \ -9.80665]$  m/s<sup>2</sup>.

**Linearization Delta**

Perturbation value for computing numerical partial derivatives during linearization. The default value is 0.001.

**Ports**

Port	Description
C	Frame port that identifies the target mechanism to which the block parameters apply.
g	Physical signal port through which you specify a time-varying gravity vector.

**See Also**

Gravitational Field

# Pin Slot Joint

Joint with one prismatic and one revolute primitives possessing mutually orthogonal motion axes

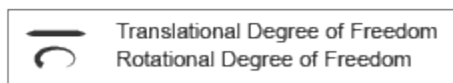
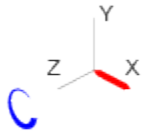
## Library

Joints



## Description

This block represents a joint with one translational and one rotational degrees of freedom. One prismatic primitive provides the translational degree of freedom. One revolute primitive provides the rotational degree of freedom. Prismatic and revolute axes are mutually orthogonal.



## Joint Degrees of Freedom

The joint block represents motion between the base and follower frames as a sequence of time-varying transformations. Each joint primitive applies one transformation in this sequence. The transformation translates or rotates the follower frame with respect to the joint primitive base frame. For all but the first joint primitive, the base frame coincides with the follower frame of the previous joint primitive in the sequence.

At each time step during the simulation, the joint block applies the sequence of time-varying frame transformations in this order:

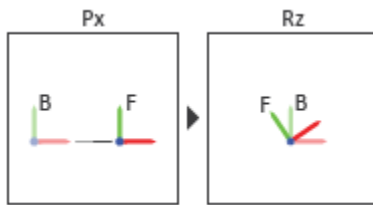
**1** Translation:

- Along the X axis of the X Prismatic Primitive (Px) base frame.

**2** Rotation:

- About the Z axis of the Z Revolute Primitive (Rz) base frame. This frame is coincident with the X Prismatic Primitive (Px) follower frame.

The figure shows the sequence in which the joint transformations occur at a given simulation time step. The resulting frame of each transformation serves as the base frame for the following transformation.



### Joint Transformation Sequence

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

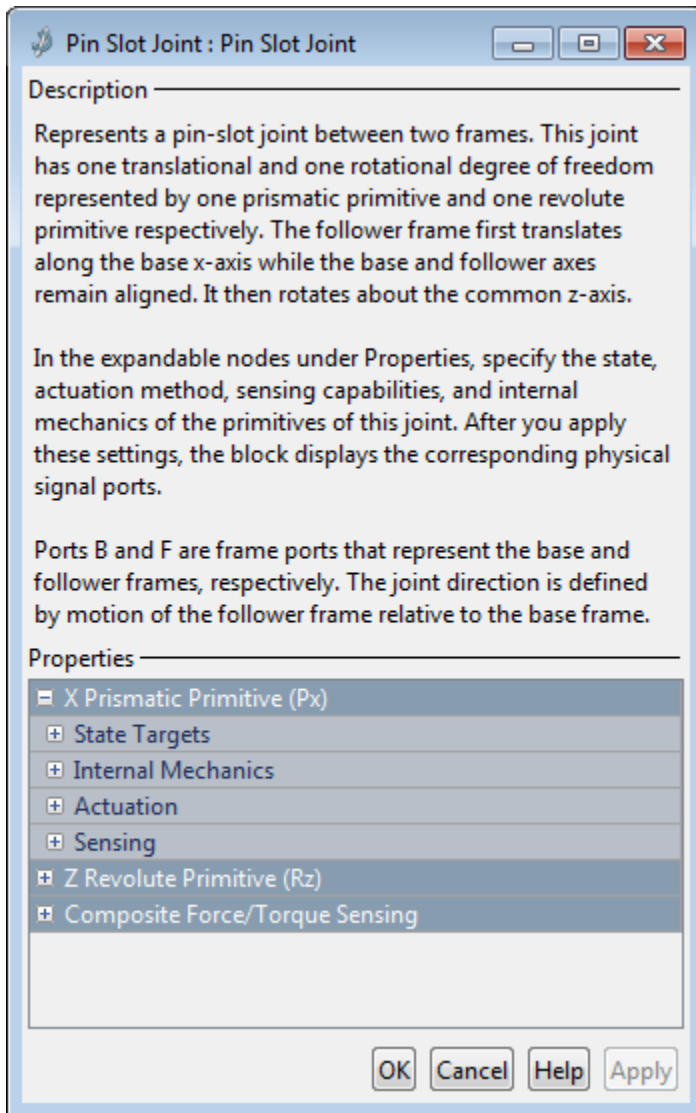
Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types: Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.



## Prismatic Primitive: State Targets

Specify the prismatic primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The



priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative position, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative velocity, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is m for position and m/s for velocity.

## Prismatic Primitive: Internal Mechanics

Specify the prismatic primitive internal mechanics. Internal mechanics include linear spring forces, accounting for energy storage, and damping forces, accounting for energy

dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

**Equilibrium Position**

Enter the spring equilibrium position. This is the distance between base and follower frame origins at which the spring force is zero. The default value is 0. Select or enter a physical unit. The default is m.

**Spring Stiffness**

Enter the linear spring constant. This is the force required to displace the joint primitive by a unit distance. The default is 0. Select or enter a physical unit. The default is N/m.

**Damping Coefficient**

Enter the linear damping coefficient. This is the force required to maintain a constant joint primitive velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is N / (m/s).

**Prismatic Primitive: Actuation**

Specify actuation options for the prismatic joint primitive. Actuation modes include **Force** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Actuation signals are resolved in the base frame.

**Force**

Select an actuation force setting. The default setting is **None**.

Actuation Force Setting	Description
None	No actuation force.
Provided by Input	Actuation force from physical signal input. The signal provides the force acting on the follower frame with respect to the base frame along the joint primitive axis. An equal and opposite force acts on the base frame.
Automatically computed	Actuation force from automatic calculation. SimMechanics computes and applies the actuation force based on model dynamics.

**Motion**

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

**Prismatic Primitive: Sensing**

Select the variables to sense in the prismatic joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

**Position**

Select this option to sense the relative position of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Velocity**

Select this option to sense the relative velocity of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Acceleration**

Select this option to sense the relative acceleration of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Actuator Force**

Select this option to sense the actuation force acting on the follower frame with respect to the base frame along the joint primitive axis.

## Revolute Primitive: State Targets

Specify the revolute primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative rotation angle, measured about the joint primitive axis, of the follower frame with respect to the base frame. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative angular velocity, measured about the joint primitive axis, of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is **deg** for position and **deg/s** for velocity.

## Revolute Primitive: Internal Mechanics

Specify the revolute primitive internal mechanics. Internal mechanics include linear spring torques, accounting for energy storage, and linear damping torques, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

### Equilibrium Position

Enter the spring equilibrium position. This is the rotation angle between base and follower frames at which the spring torque is zero. The default value is 0. Select or enter a physical unit. The default is **deg**.

### Spring Stiffness

Enter the linear spring constant. This is the torque required to rotate the joint primitive by a unit angle. The default is 0. Select or enter a physical unit. The default is **N\*m/deg**.

### Damping Coefficient

Enter the linear damping coefficient. This is the torque required to maintain a constant joint primitive angular velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is **N\*m/(deg/s)**.

## Revolute Primitive: Actuation

Specify actuation options for the revolute joint primitive. Actuation modes include **Torque** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Input signals are resolved in the base frame.

### Torque

Select an actuation torque setting. The default setting is **None**.

Actuation Torque Setting	Description
None	No actuation torque.
Provided by Input	Actuation torque from physical signal input. The signal provides the torque acting on the follower frame with respect to the base frame about the joint primitive axis. An equal and opposite torque acts on the base frame.

Actuation Torque Setting	Description
Automatically computed	Actuation torque from automatic calculation. SimMechanics computes and applies the actuation torque based on model dynamics.

### Motion

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

### Revolute Primitive: Sensing

Select the variables to sense in the revolute joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

#### Position

Select this option to sense the relative rotation angle of the follower frame with respect to the base frame about the joint primitive axis.

#### Velocity

Select this option to sense the relative angular velocity of the follower frame with respect to the base frame about the joint primitive axis.

#### Acceleration

Select this option to sense the relative angular acceleration of the follower frame with respect to the base frame about the joint primitive axis.

### **Actuator Torque**

Select this option to sense the actuation torque acting on the follower frame with respect to the base frame about the joint primitive axis.

## **Composite Force/Torque Sensing**

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

### **Direction**

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

### **Resolution Frame**

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

### **Constraint Force**

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint

force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector [ $f_{c_x}$ ,  $f_{c_y}$ ,  $f_{c_z}$ ] acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

### **Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector [ $tc_x$ ,  $tc_y$ ,  $tc_z$ ] acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

### **Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector [ $ft_x$ ,  $ft_y$ ,  $ft_z$ ]. Selecting this option causes the block to expose physical signal port *ft*.

### **Total Torque**

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.
- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.



The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port  $tt$ .

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
f	Force	Actuation input	Scalar
t	Torque	Actuation input	Scalar
p	Position	Sensing output	Scalar
v	Velocity	Sensing output	Scalar
a	Acceleration	Sensing output	Scalar
q	Rotation angle	Sensing output	Scalar
w	Angular velocity	Sensing output	Scalar
b	Angular acceleration	Sensing output	Scalar

## See Also

Cylindrical Joint | Revolute joint | Prismatic Joint

## More About

- “Actuating and Sensing Using Physical Signals”
- “Motion Sensing”

- “Rotational Measurements”
- “Translational Measurements”

# Planar Joint

Joint with one revolute and two prismatic primitives

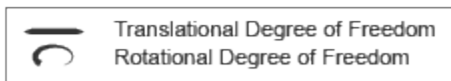
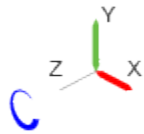
## Library

Joints



## Description

This block represents a joint with one rotational and two translational degrees of freedom. Two prismatic primitives provide the two translational degrees of freedom. One revolute primitive provides the rotational degree of freedom.



## Joint Degrees of Freedom

The joint block represents motion between the base and follower frames as a sequence of time-varying transformations. Each joint primitive applies one transformation in this sequence. The transformation translates or rotates the follower frame with respect to the joint primitive base frame. For all but the first joint primitive, the base frame coincides with the follower frame of the previous joint primitive in the sequence.

At each time step during the simulation, the joint block applies the sequence of time-varying frame transformations in this order:

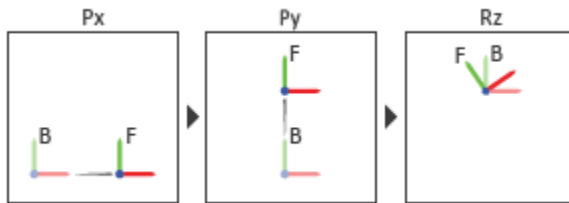
## 1 Translation:

- a** Along the X axis of the X Prismatic Primitive (Px) base frame.
- b** Along the Y axis of the Y Prismatic Primitive (Py) base frame. This frame is coincident with the X Prismatic Primitive (Px) follower frame.

## 2 Rotation:

- About the Z axis of the Z Revolute Primitive (Rz) base frame. This frame is coincident with the Y Prismatic Primitive (Py) follower frame.

The figure shows the sequence in which the joint transformations occur at a given simulation time step. The resulting frame of each transformation serves as the base frame for the following transformation.



### Joint Transformation Sequence

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

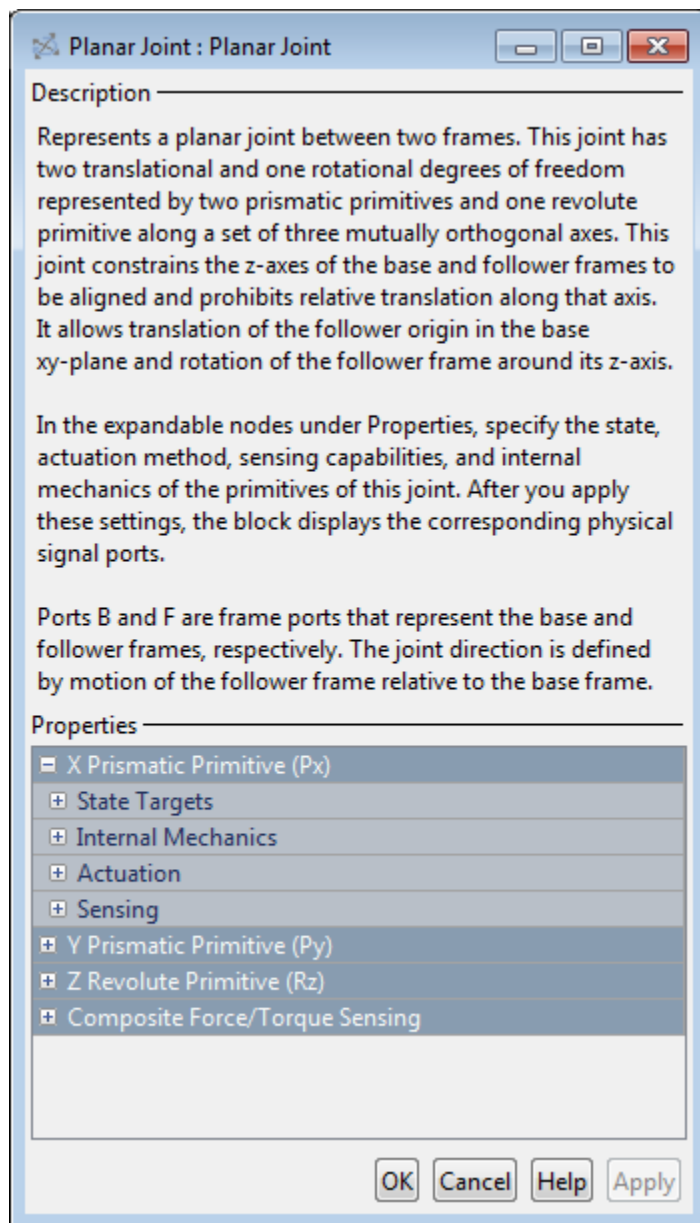
Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types: Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.



## Prismatic Primitive: State Targets

Specify the prismatic primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative position, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative velocity, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is m for position and m/s for velocity.

## Prismatic Primitive: Internal Mechanics

Specify the prismatic primitive internal mechanics. Internal mechanics include linear spring forces, accounting for energy storage, and damping forces, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

### Equilibrium Position

Enter the spring equilibrium position. This is the distance between base and follower frame origins at which the spring force is zero. The default value is 0. Select or enter a physical unit. The default is m.

### Spring Stiffness

Enter the linear spring constant. This is the force required to displace the joint primitive by a unit distance. The default is 0. Select or enter a physical unit. The default is N/m.

### Damping Coefficient

Enter the linear damping coefficient. This is the force required to maintain a constant joint primitive velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is N / (m/s).

## Prismatic Primitive: Actuation

Specify actuation options for the prismatic joint primitive. Actuation modes include **Force** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Actuation signals are resolved in the base frame.

### Force

Select an actuation force setting. The default setting is **None**.

Actuation Force Setting	Description
None	No actuation force.
Provided by Input	Actuation force from physical signal input. The signal provides the force acting on the follower frame with respect to the base frame along the joint primitive axis. An equal and opposite force acts on the base frame.



Actuation Force Setting	Description
Automatically computed	Actuation force from automatic calculation. SimMechanics computes and applies the actuation force based on model dynamics.

### Motion

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

### Prismatic Primitive: Sensing

Select the variables to sense in the prismatic joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

#### Position

Select this option to sense the relative position of the follower frame origin with respect to the base frame origin along the joint primitive axis.

#### Velocity

Select this option to sense the relative velocity of the follower frame origin with respect to the base frame origin along the joint primitive axis.

#### Acceleration

Select this option to sense the relative acceleration of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Actuator Force**

Select this option to sense the actuation force acting on the follower frame with respect to the base frame along the joint primitive axis.

**Revolute Primitive: State Targets**

Specify the revolute primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

**Specify Position Target**

Select this option to specify the desired joint primitive position at time zero. This is the relative rotation angle, measured about the joint primitive axis, of the follower frame with respect to the base frame. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

**Specify Velocity Target**

Select this option to specify the desired joint primitive velocity at time zero. This is the relative angular velocity, measured about the joint primitive axis, of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields.

**Priority**

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

**Value**

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is **deg** for position and **deg/s** for velocity.

**Revolute Primitive: Internal Mechanics**

Specify the revolute primitive internal mechanics. Internal mechanics include linear spring torques, accounting for energy storage, and linear damping torques, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

**Equilibrium Position**

Enter the spring equilibrium position. This is the rotation angle between base and follower frames at which the spring torque is zero. The default value is 0. Select or enter a physical unit. The default is **deg**.

**Spring Stiffness**

Enter the linear spring constant. This is the torque required to rotate the joint primitive by a unit angle. The default is 0. Select or enter a physical unit. The default is **N\*m/deg**.

**Damping Coefficient**

Enter the linear damping coefficient. This is the torque required to maintain a constant joint primitive angular velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is **N\*m/(deg/s)**.

**Revolute Primitive: Actuation**

Specify actuation options for the revolute joint primitive. Actuation modes include **Torque** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Input signals are resolved in the base frame.

**Torque**

Select an actuation torque setting. The default setting is **None**.

Actuation Torque Setting	Description
None	No actuation torque.

Actuation Torque Setting	Description
Provided by Input	Actuation torque from physical signal input. The signal provides the torque acting on the follower frame with respect to the base frame about the joint primitive axis. An equal and opposite torque acts on the base frame.
Automatically computed	Actuation torque from automatic calculation. SimMechanics computes and applies the actuation torque based on model dynamics.

### Motion

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

### Revolute Primitive: Sensing

Select the variables to sense in the revolute joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

### Position

Select this option to sense the relative rotation angle of the follower frame with respect to the base frame about the joint primitive axis.

**Velocity**

Select this option to sense the relative angular velocity of the follower frame with respect to the base frame about the joint primitive axis.

**Acceleration**

Select this option to sense the relative angular acceleration of the follower frame with respect to the base frame about the joint primitive axis.

**Actuator Torque**

Select this option to sense the actuation torque acting on the follower frame with respect to the base frame about the joint primitive axis.

**Composite Force/Torque Sensing**

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

**Direction**

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

**Resolution Frame**

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

**Constraint Force**

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector  $[fc_x, fc_y, fc_z]$  acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

**Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector  $[tc_x, tc_y, tc_z]$  acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

**Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector  $[ft_x, ft_y, ft_z]$ . Selecting this option causes the block to expose physical signal port *ft*.

**Total Torque**

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.

- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port  $tt$ .

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
f	Force	Actuation input	Scalar
t	Torque	Actuation input	Scalar
p	Position	Sensing output	Scalar
v	Velocity	Sensing output	Scalar
a	Acceleration	Sensing output	Scalar
q	Rotation angle	Sensing output	Scalar
w	Angular velocity	Sensing output	Scalar
b	Angular acceleration	Sensing output	Scalar

## See Also

Rectangular Joint | Prismatic Joint | Revolute Joint

## **More About**

- “Actuating and Sensing Using Physical Signals”
- “Motion Sensing”
- “Rotational Measurements”
- “Translational Measurements”

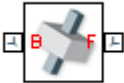


# Prismatic Joint

Joint with one prismatic primitive

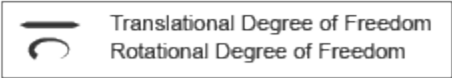
## Library

Joints



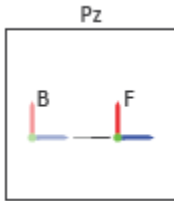
## Description

This block represents a joint with one translational degree of freedom. One prismatic primitive provides the translational degree of freedom. The base and follower frames remain parallel during simulation.



## Joint Degrees of Freedom

The joint block represents motion between the base and follower frames as a single time-varying transformation. The Z prismatic primitive (Pz) applies this transformation, which causes the follower frame to translate with respect to the base frame along the common Z axis.



## Joint Transformation

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

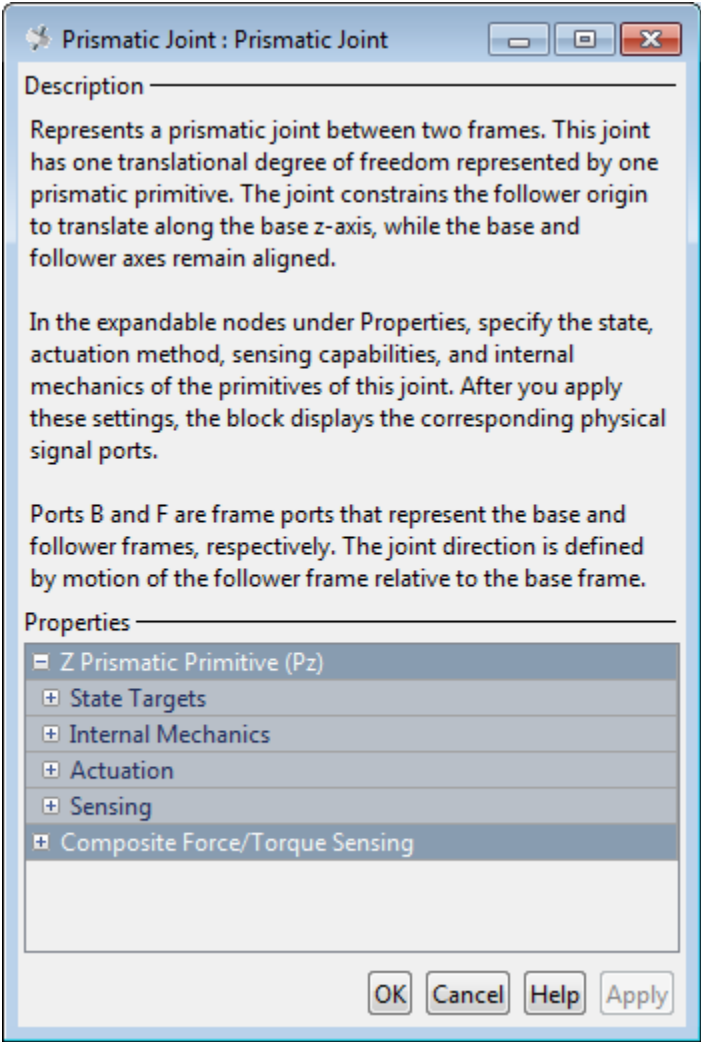
Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types: Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.



### Prismatic Primitive: State Targets

Specify the prismatic primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely

the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative position, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative velocity, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is m for position and m/s for velocity.

## Prismatic Primitive: Internal Mechanics

Specify the prismatic primitive internal mechanics. Internal mechanics include linear spring forces, accounting for energy storage, and damping forces, accounting for energy

dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

**Equilibrium Position**

Enter the spring equilibrium position. This is the distance between base and follower frame origins at which the spring force is zero. The default value is 0. Select or enter a physical unit. The default is m.

**Spring Stiffness**

Enter the linear spring constant. This is the force required to displace the joint primitive by a unit distance. The default is 0. Select or enter a physical unit. The default is N/m.

**Damping Coefficient**

Enter the linear damping coefficient. This is the force required to maintain a constant joint primitive velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is N/ (m/s).

**Prismatic Primitive: Actuation**

Specify actuation options for the prismatic joint primitive. Actuation modes include **Force** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Actuation signals are resolved in the base frame.

**Force**

Select an actuation force setting. The default setting is **None**.

Actuation Force Setting	Description
None	No actuation force.
Provided by Input	Actuation force from physical signal input. The signal provides the force acting on the follower frame with respect to the base frame along the joint primitive axis. An equal and opposite force acts on the base frame.
Automatically computed	Actuation force from automatic calculation. SimMechanics computes and applies the actuation force based on model dynamics.

**Motion**

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

**Prismatic Primitive: Sensing**

Select the variables to sense in the prismatic joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

**Position**

Select this option to sense the relative position of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Velocity**

Select this option to sense the relative velocity of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Acceleration**

Select this option to sense the relative acceleration of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Actuator Force**

Select this option to sense the actuation force acting on the follower frame with respect to the base frame along the joint primitive axis.

## Composite Force/Torque Sensing

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

### Direction

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

### Resolution Frame

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

### Constraint Force

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector  $[f_{c_x}, f_{c_y}, f_{c_z}]$  acting at the joint. Only constraint force components that are orthogonal to the

joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

**Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector [ $tc_x$ ,  $tc_y$ ,  $tc_z$ ] acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

**Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector [ $ft_x$ ,  $ft_y$ ,  $ft_z$ ]. Selecting this option causes the block to expose physical signal port *ft*.

**Total Torque**

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.
- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector [ $tt_x$ ,  $tt_y$ ,  $tt_z$ ]. Selecting this option causes the block to expose physical signal port *tt*.



## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Input/Output
f	Force	Scalar
p	Position	Scalar
v	Velocity	Scalar
a	Acceleration	Scalar

### See Also

Revolute Joint | Spherical Joint

### More About

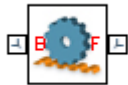
- “Actuating and Sensing Using Physical Signals”
- “Motion Sensing”
- “Translational Measurements”

# Rack and Pinion Constraint

Kinematic constraint for converting between rotation and translation

## Library

Gears and Couplings/Gears



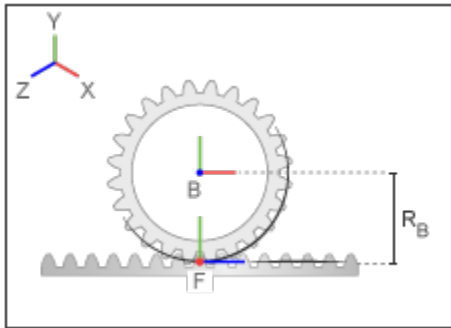
## Description

This block represents a kinematic constraint between a rack and a pinion. The constraint converts rotation of the pinion into translation of the rack and vice-versa. The pinion, which connects to the base port frame, spins about the base Z axis. The rack, which connects to the follower port frame, translates along the follower Z axis.

Kinematic constraints in the remainder of the model must hold the base and follower frames at the correct distance and with the proper alignment. These constraints might be due to rigid transforms, joints, and model topology. Assembly conditions include:

- Base and follower Z axes must be mutually orthogonal. This condition ensures that the pinion rotation axis sits at a right angle to the rack translation axis. You can rotate frames using the Rigid Transform block.
- Base and follower frame origins must be apart by a distance equal to the **Pinion Radius** parameter. This condition ensures that the rack and pinion cogs are at the correct distance for engagement. You can translate frames using the Rigid Transform block.

The figure shows the distance and alignment of the base and follower frames in the zero configuration. This is the primary configuration that SimMechanics attempts to achieve during model assembly.



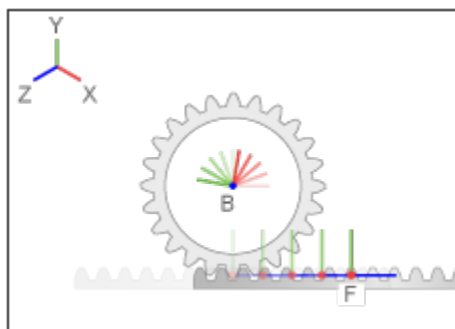
In the zero configuration, the pinion rotation angle and the rack translation distance are both zero. To achieve this configuration, SimMechanics:

- Aligns the base and follower Y axes.
- Positions the follower frame origin along the negative Y axis of the base frame.

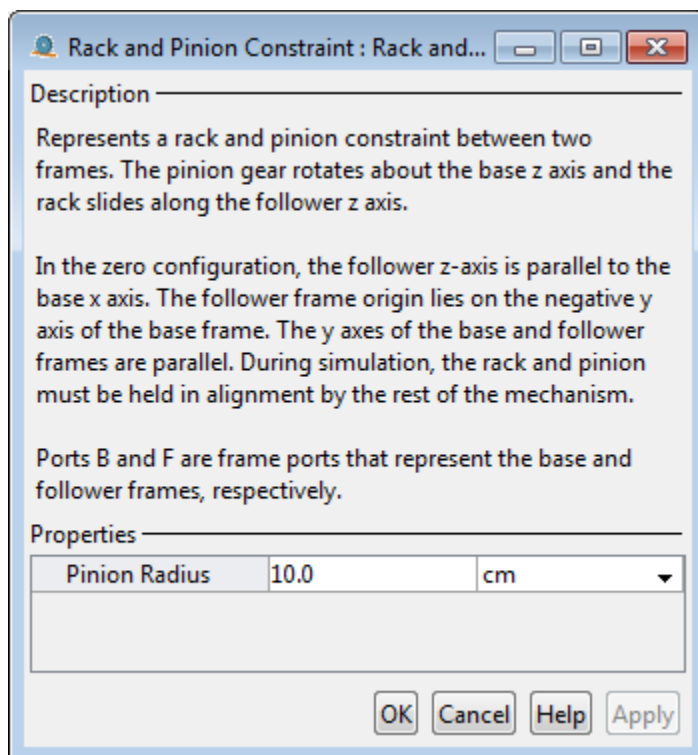
Joint blocks provide the base and follower frames with the proper internal degrees of freedom. These degrees of freedom must support rotation about the base Z axis and translation along the follower Z axis. You can achieve these degrees of freedom using different joint block combinations. For example, you can connect the pinion to a Revolute Joint block and the rack to a Prismatic Joint block.

During simulation, a positive pinion rotation about the base Z axis corresponds to a positive rack translation along the follower Z axis. By definition, the translational velocity of the rack is equal to the tangential velocity at a point in the pinion pitch circle, an imaginary circle that intersects the rack and pinion cogs at the mutual contact point.

The figure shows the relative motion of the base and follower frames due to the rack and pinion constraint.



## Dialog Box and Parameters



### **Pinion Radius**

Distance between the pinion center and pitch circle. This circle contains the instantaneous contact point between a pair of rack and pinion cogs. The pinion radius must equal the actual distance between the base and follower frames as specified by the remainder of the model. The default value is 10 cm.

### **Ports**

The block contains frame ports B and F, representing base and follower frames, respectively.

### **See Also**

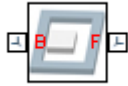
Common Gear Constraint | Bevel Gear Constraint

# Rectangular Joint

Joint with two prismatic primitives

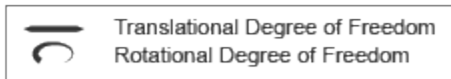
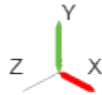
## Library

Joints



## Description

This block represents a joint with two translational degrees of freedom. Two prismatic primitives provide the two translational degrees of freedom. The base and follower frames remain parallel during simulation.



## Joint Degrees of Freedom

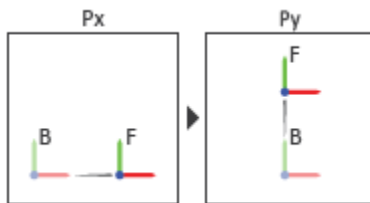
The joint block represents motion between the base and follower frames as a sequence of time-varying transformations. Each joint primitive applies one transformation in this sequence. The transformation translates or rotates the follower frame with respect to the joint primitive base frame. For all but the first joint primitive, the base frame coincides with the follower frame of the previous joint primitive in the sequence.

At each time step during the simulation, the joint block applies the sequence of time-varying frame transformations in this order:

## 1 Translation:

- a Along the X axis of the X Prismatic Primitive (Px) base frame.
- b Along the Y axis of the Y Prismatic Primitive (Py) base frame. This frame is coincident with the X Prismatic Primitive (Px) follower frame.

The figure shows the sequence in which the joint transformations occur at a given simulation time step. The resulting frame of each transformation serves as the base frame for the following transformation.



### Joint Transformation Sequence

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

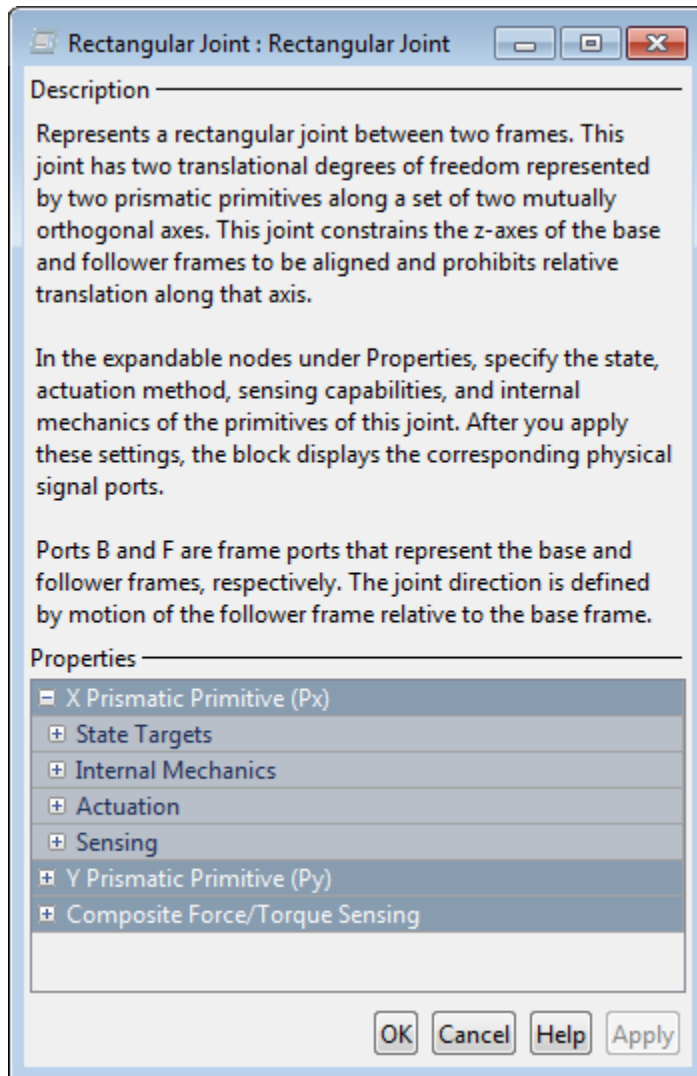
## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types:

Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.





## Prismatic Primitive: State Targets

Specify the prismatic primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely

the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative position, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative velocity, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is m for position and m/s for velocity.

## Prismatic Primitive: Internal Mechanics

Specify the prismatic primitive internal mechanics. Internal mechanics include linear spring forces, accounting for energy storage, and damping forces, accounting for energy

dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

### Equilibrium Position

Enter the spring equilibrium position. This is the distance between base and follower frame origins at which the spring force is zero. The default value is 0. Select or enter a physical unit. The default is m.

### Spring Stiffness

Enter the linear spring constant. This is the force required to displace the joint primitive by a unit distance. The default is 0. Select or enter a physical unit. The default is N/m.

### Damping Coefficient

Enter the linear damping coefficient. This is the force required to maintain a constant joint primitive velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is N/(m/s).

## Prismatic Primitive: Actuation

Specify actuation options for the prismatic joint primitive. Actuation modes include **Force** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Actuation signals are resolved in the base frame.

### Force

Select an actuation force setting. The default setting is **None**.

Actuation Force Setting	Description
None	No actuation force.
Provided by Input	Actuation force from physical signal input. The signal provides the force acting on the follower frame with respect to the base frame along the joint primitive axis. An equal and opposite force acts on the base frame.
Automatically computed	Actuation force from automatic calculation. SimMechanics computes and applies the actuation force based on model dynamics.

**Motion**

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

**Prismatic Primitive: Sensing**

Select the variables to sense in the prismatic joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

**Position**

Select this option to sense the relative position of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Velocity**

Select this option to sense the relative velocity of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Acceleration**

Select this option to sense the relative acceleration of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Actuator Force**

Select this option to sense the actuation force acting on the follower frame with respect to the base frame along the joint primitive axis.

## Composite Force/Torque Sensing

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

### Direction

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

### Resolution Frame

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

### Constraint Force

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector  $[f_{c_x}, f_{c_y}, f_{c_z}]$  acting at the joint. Only constraint force components that are orthogonal to the

joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

### **Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector  $[tc_x, tc_y, tc_z]$  acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

### **Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector  $[ft_x, ft_y, ft_z]$ . Selecting this option causes the block to expose physical signal port *ft*.

### **Total Torque**

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.
- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port *tt*.

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
f	Force	Actuation input	Scalar
p	Position	Sensing output	Scalar
v	Velocity	Sensing output	Scalar
a	Acceleration	Sensing output	Scalar

## See Also

Prismatic Joint | Planar Joint

## More About

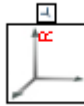
- “Actuating and Sensing Using Physical Signals”
- “Motion Sensing”
- “Translational Measurements”

## Reference Frame

Non-inertial reference frame

## Library

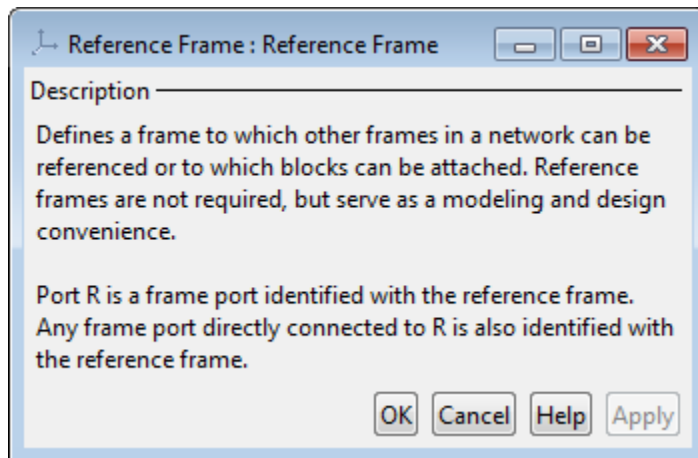
Frames and Transforms



## Description

This block represents a reference frame with respect to which you can define other frames. The reference frame is generally non-inertial. It can accelerate with respect to the World frame. This block is optional in a model.

## Dialog Box and Parameters





## Ports

This block contains frame port  $R$ , representing the reference frame.

## See Also

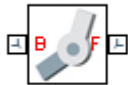
World Frame | Rigid Transform

# Revolute Joint

Joint with one revolute primitive

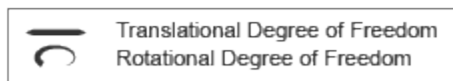
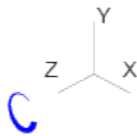
## Library

Joints



## Description

This block represents a joint with one rotational degree of freedom. One revolute primitive provides the rotational degree of freedom. The base and follower frame origins remain coincident during simulation.



## Joint Degrees of Freedom

The joint block represents motion between the base and follower frames as a single time-varying transformation. The Z revolute primitive (Pz) applies this transformation, which causes the follower frame to rotate with respect to the base frame about the common Z axis.



### Joint Transformation

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

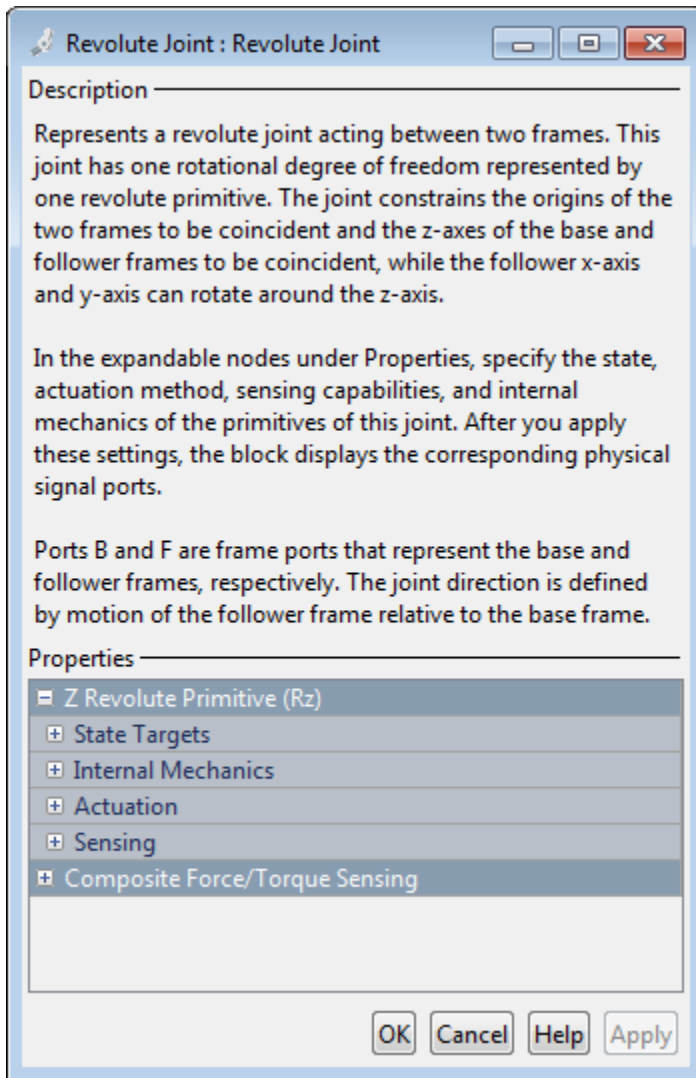
Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types: Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.



## Revolute Primitive: State Targets

Specify the revolute primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target

must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative rotation angle, measured about the joint primitive axis, of the follower frame with respect to the base frame. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative angular velocity, measured about the joint primitive axis, of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is **deg** for position and **deg/s** for velocity.

## Revolute Primitive: Internal Mechanics

Specify the revolute primitive internal mechanics. Internal mechanics include linear spring torques, accounting for energy storage, and linear damping torques, accounting

for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

**Equilibrium Position**

Enter the spring equilibrium position. This is the rotation angle between base and follower frames at which the spring torque is zero. The default value is 0. Select or enter a physical unit. The default is deg.

**Spring Stiffness**

Enter the linear spring constant. This is the torque required to rotate the joint primitive by a unit angle. The default is 0. Select or enter a physical unit. The default is N\*m/deg.

**Damping Coefficient**

Enter the linear damping coefficient. This is the torque required to maintain a constant joint primitive angular velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is N\*m/ (deg/s).

**Revolute Primitive: Actuation**

Specify actuation options for the revolute joint primitive. Actuation modes include **Torque** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Input signals are resolved in the base frame.

**Torque**

Select an actuation torque setting. The default setting is **None**.

Actuation Torque Setting	Description
None	No actuation torque.
Provided by Input	Actuation torque from physical signal input. The signal provides the torque acting on the follower frame with respect to the base frame about the joint primitive axis. An equal and opposite torque acts on the base frame.
Automatically computed	Actuation torque from automatic calculation. SimMechanics computes and applies the actuation torque based on model dynamics.

**Motion**

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

**Revolute Primitive: Sensing**

Select the variables to sense in the revolute joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

**Position**

Select this option to sense the relative rotation angle of the follower frame with respect to the base frame about the joint primitive axis.

**Velocity**

Select this option to sense the relative angular velocity of the follower frame with respect to the base frame about the joint primitive axis.

**Acceleration**

Select this option to sense the relative angular acceleration of the follower frame with respect to the base frame about the joint primitive axis.

**Actuator Torque**

Select this option to sense the actuation torque acting on the follower frame with respect to the base frame about the joint primitive axis.

## Composite Force/Torque Sensing

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

### Direction

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

### Resolution Frame

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

### Constraint Force

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector  $[f_{c_x}, f_{c_y}, f_{c_z}]$  acting at the joint. Only constraint force components that are orthogonal to the



joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

### Constraint Torque

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector  $[tc_x, tc_y, tc_z]$  acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

### Total Force

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector  $[ft_x, ft_y, ft_z]$ . Selecting this option causes the block to expose physical signal port *ft*.

### Total Torque

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.
- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port *tt*.

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
t	Torque	Actuation input	Scalar
q	Rotation angle	Sensing output	Scalar
w	Angular velocity	Sensing output	Scalar
b	Angular acceleration	Sensing output	Scalar

## See Also

Prismatic Joint | Spherical Joint

## More About

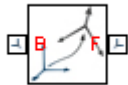
- “Actuating and Sensing Using Physical Signals”
- “Motion Sensing”
- “Rotational Measurements”

# Rigid Transform

Time-invariant rotation and translation transforms between two frames

## Library

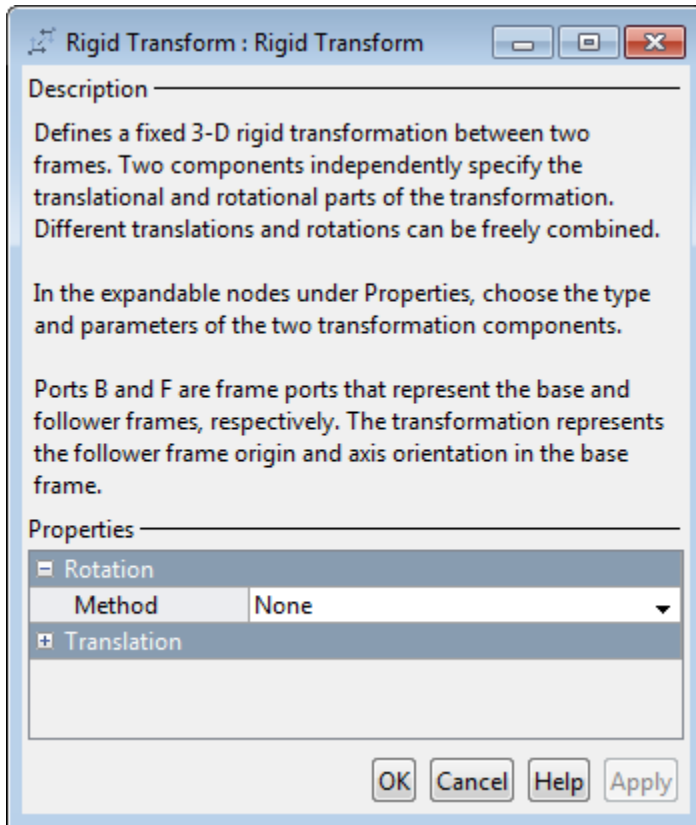
Frames and Transforms



## Description

This block applies a time-invariant transformation between two frames. The transformation rotates and translates the follower port frame (F) with respect to the base port frame (B). Connecting the frame ports in reverse causes the transformation itself to reverse. The frames remain fixed with respect to each other during simulation, moving only as a single unit. Combine Rigid Transform and Solid blocks to model compound rigid bodies.

## Dialog Box and Parameters



### Rotation:Method

Select the method to use to specify rotation. The default is None.

Method	Description
None	Align base and follower frames.
Aligned Axes	Align two follower frame axes with two base frame axes.

Method	Description
Standard Axis	Specify a rotation angle about a standard axis ( $x$ , $y$ , or $z$ ).
Arbitrary Axis	Specify a rotation angle about a general $[x, y, z]$ axis.
Rotation Matrix	Specify a right-handed orthogonal rotation matrix.

### Aligned Axes

Select two pairs of base-follower frame axes.

Parameter	Description
Pair 1	First pair of base-follower frame axes to align.
Pair 2	Second pair of base-follower frame axes to align. Axis choices depend on <b>Pair 1</b> axis selections.

### Standard Axis

Select a standard rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
Axis	Standard rotation axis (X, Y, or Z) resolved in the base frame.
Angle	Follower frame rotation angle about the rotation axis with respect to the base frame.

### Arbitrary Axis

Select a general 3-D rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
Axis	General rotation axis $[X Y Z]$ resolved in the base frame.

Parameter	Description
Angle	Follower frame rotation angle about the rotation axis with respect to the base frame.

### Rotation Matrix

Specify the 3×3 transformation matrix of a proper rotation between the base and follower frames. The matrix must be orthogonal and have determinant +1. The default value is [1 0 0; 0 1 0; 0 0 1].

### Translation: Method

Select the method to use to specify translation. The default is None.

Method	Description
None	Make base and follower frames coincident. This method requires no parameters.
Cartesian	Specify a 3-D translation in terms of Cartesian coordinates
Standard Axis	Specify a 1-D translation along the X, Y, or Z axis
Cylindrical	Specify a 3-D translation in terms of cylindrical coordinates

### Cartesian Axis

Specify the **Offset** of the follower frame with respect to the base frame. This is the 3-D translation vector that brings the base frame into coincidence with the follower frame. Select or enter a physical unit.

### Standard Axis

Specify the offset of the follower frame with respect to the base frame along the base frame X, Y, or Z axis. Select or enter a physical unit.

Parameter	Description
Axis	Axis the follower frame translates along

Parameter	Description
<b>Offset</b>	Translation of the follower frame with respect to the base frame along the specified axis

### Cylindrical

Specify in cylindrical coordinates the translation that brings the base frame into coincidence with the follower frame. Select or enter a physical unit.

Parameter	Description
<b>Radius</b>	Distance between the origin of the follower frame and the Z axis of the base frame. This is the cylindrical radius coordinate.
<b>Theta</b>	Rotation angle of the line connecting base and follower frame origins with respect to the base frame X axis. This is the cylindrical azimuth coordinate.
<b>Z Offset</b>	Distance between base and follower frame origins along the base frame Z axis. This is the cylindrical length coordinate.

## Ports

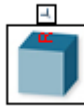
Frame ports B and F represent the base and follower frames.

## Solid

Rigid solid with geometry, inertia, and color

## Library

Body Elements



## Description

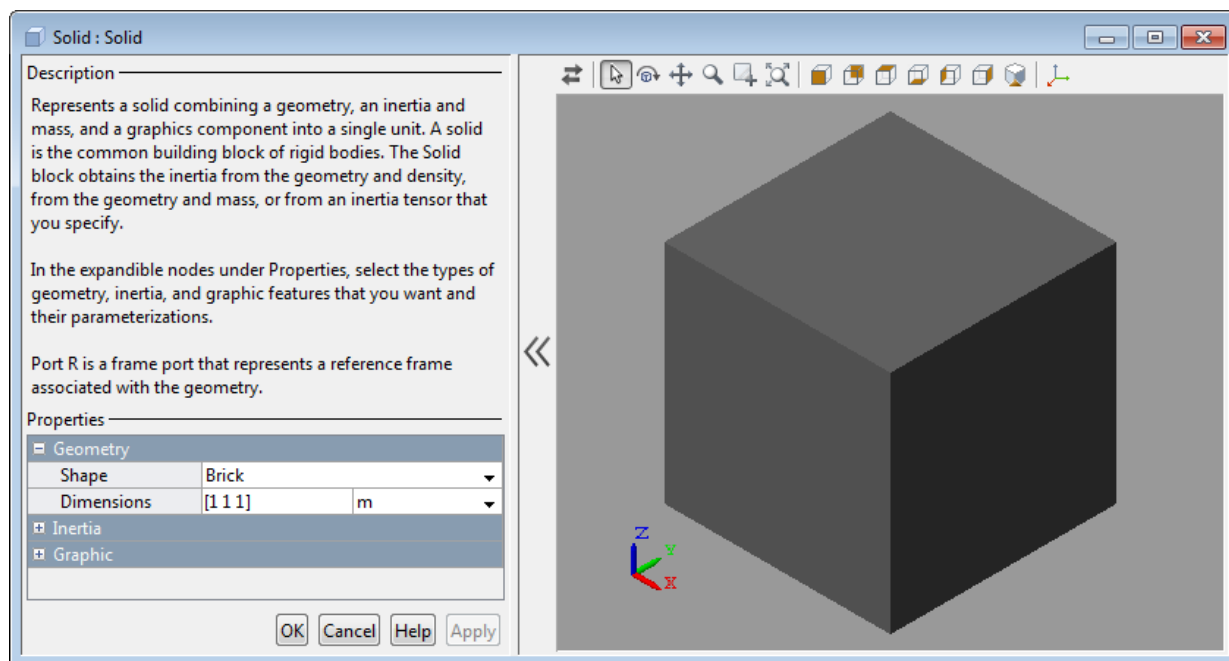
This block represents a rigid solid with geometry, inertia, and color. The solid can be a simple rigid body or part of a compound rigid body—a group of rigidly connected solids, often separated in space through rigid transformations. Combine Solid and Rigid Transform blocks to model a compound rigid body.

Geometry parameters include shape and size. You can choose from a list of preset shapes or import a custom shape from an external file in STL or STEP format. The block can automatically compute the inertial properties of the solid based on the geometry you specify given its mass or mass density.

The block dialog box contains a collapsible visualization pane. This pane provides instant visual feedback on the solid you are modeling. Use it to find and fix any issues with the shape and color of the solid. You can examine the solid from different perspectives by selecting a standard view or by rotating, panning, and zooming the solid.




## Dialog Box and Parameters









## Geometry

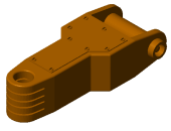
### Shape

Select a solid shape. The table summarizes the various shapes that you can select. The default shape is Brick.

Shape	Description	Example
Cylinder	Cylindrical shape with geometry center at the reference frame origin and symmetry axis aligned with reference frame Z axis	

Shape	Description	Example
Sphere	Spherical shape with geometry center at the reference frame origin.	
Brick	Prismatic shape with geometry center at the reference frame origin and faces normal to X, Y, Z axes.	
Ellipsoid	3-D extension of ellipse with geometry center at the reference frame origin and semi-principal axes aligned with reference frame X, Y, Z axes.	
Regular Extrusion	<p>3-D sweep of regular polygon cross-section along an extrusion axis.</p> <p>Shape has geometry center at the reference frame origin, and extrusion axis aligned with reference frame Z axis. Cross-section is constant along extrusion length.</p>	

Shape	Description	Example
General Extrusion	<p>3-D sweep of general cross-section shape along an extrusion axis.</p> <p>Reference frame origin coincides with cross-section (0,0) coordinate, halfway along extrusion length. Reference frame Z axis aligns with extrusion axis.</p> <p>Cross-section lies in reference frame XY plane. Cross-section shape and dimensions are constant along extrusion length.</p>	
Revolution	<p>3-D sweep of general cross-section about a revolution axis.</p> <p>Reference frame origin coincides with cross-section (0,0) coordinate. Reference frame Z axis aligns with revolution axis.</p> <p>Cross-section lies in reference frame XZ plane. Revolutions can be full (revolution angle = <math>360^\circ</math>) or partial (<math>0^\circ &lt; \text{revolution angle} &lt; 360^\circ</math>). For partial revolutions, the reference frame X axis splits the revolution into two symmetric halves.</p>	

Shape	Description	Example
From File	<p>3-D shape loaded from STL (Standard Tessellation Language) or STEP (Standard for the Exchange of Product Data) file.</p> <p>The reference frame has the origin and orientation defined in the file.</p>	

**Cylinder: Radius**

Enter the cylinder radius. This is the distance between the origin and circumference of the transverse cross-section. The default value is 1. Select or enter a physical unit. The default is m.

**Cylinder: Length**

Enter the cylinder length. This is the distance between the two flat surfaces measured along the symmetry axis. The default value is 1. Select or enter a physical unit. The default is m.

**Sphere: Radius**

Enter the spherical radius. This is the distance between the origin and surface of the sphere. The default value is 1.

**Brick: Dimensions**

Enter a three element vector  $[a \ b \ c]$  with the brick dimensions along the reference frame X, Y, and Z axes, respectively. The default vector is  $[1 \ 1 \ 1]$ . Select a physical unit. The default unit is m.

**Ellipsoid: Radii**

Enter a three element vector  $[a \ b \ c]$  with the ellipsoid semi-principal axes along the reference frame X, Y, and Z axes, respectively. The default vector is  $[1 \ 1 \ 1]$ . Select a physical unit. The default unit is m.

**Regular Extrusion: Number of Sides**

Enter the number of sides for the polygonal cross-section. The minimum number of sides is 3. The default value is 3.

**Regular Extrusion: Outer Radius**

Enter the radius of the smallest circle required to completely enclose the polygonal cross-section. This is equal to the distance from the polygon center to the intersection of any two polygon edges. The default value is 1. Select a physical unit. The default unit is m.

**Regular Extrusion: Length**

Enter the extrusion length. This is the distance along which to sweep the 2-D cross-section. The default value is 1. Select a physical unit. The default unit is m.

**General Extrusion: Cross-section**

Enter the cross-section coordinate matrix. This is a matrix with  $N$  rows, each with the [X Y] coordinates of a single cross-section point. Coordinates must define a single closed loop. The loop must not self-intersect. The closed loop divides dense and empty regions according to the following rule: as viewed at each point along the cross-section, the dense region lies to the left of the cross-section segment, while the empty region lies to the right. Select a physical unit. The default unit is m.

**General Extrusion: Length**

Enter the extrusion length. This is the distance along which to sweep the 2-D cross-section. The default value is 1. Select a physical unit. The default unit is m.

**Revolution: Cross-section**

Enter the cross-section coordinate matrix. This is a matrix with  $N$  rows, each with the [X Z] coordinates of a single cross-section point. Coordinates must define a closed loop. The loop must not self-intersect. X-coordinate values must be greater than or equal to zero. The closed loop divides dense and empty regions according to the following rule: as viewed at each point along the cross-section, the dense region lies to the left of the cross-section segment, while the empty region lies to the right. Select a physical unit. The default unit is m.

**Revolution: Extent of Revolution**

Specify the angle to revolve the cross-section through. Select **Full** for a 360 degree revolution. Select **Custom** and enter a revolution angle for partial revolutions. The revolution angle must lie between 0 and 360 degrees.

**From File: File Type**

Select the format of the source file with the solid geometry data. Formats include STL and STEP.

STL (Standard Tessellation Language) files represent the surface geometry of a 3-D solid as a matrix of 2-D triangular elements. A normal vector and three vertex

coordinate sets, included in the STL file, fully define each triangular element in the tessellated surface. Selecting STL exposes an additional option, **Units**.

STEP (Standard for the Exchange of Product Data) files represent the surface geometry of a 3-D solid using a set of analytical curves. These files can include additional information about a solid, such as mass density and physical units.

The block provides automatic inertia computation from geometry only for STEP-derived geometries. For STL-derived geometries, you must manually enter the solid inertia parameters.

**From File: File Name**

Enter the name of the geometry source file. The name must include the file path, provided relative to the working directory.

**From File: Units**

Select or enter the desired unit of length. The default is m. This option appears when you select STL as the geometry source file type.

## Inertia

### Type

Select a method to specify the inertial properties of the solid. The default is Calculate from Geometry.

Type	Description
Calculate from Geometry	Automatically compute moments and products of inertia based on solid geometry and either mass or density.
Point Mass	Treat the solid as an idealized mass occupying an infinitely small volume in space. The inertia tensor about the center of mass is always zero for a point mass. The position of the point mass coincides with the origin of the reference port frame. Select the Point Mass method to represent a simple mass disturbance on a rigid body.

Type	Description
Custom	Manually specify the inertial properties of the solid, including moments and products of inertia as well as center of mass.

### Calculate from Geometry: **Based on**

Select the quantity to base inertia calculations on. Options are **Density** and **Mass**. Depending on the method you choose, enter the average mass density or the total mass of the solid. Select a physical unit.

### Point Mass/Custom: **Mass**

Enter the total mass of the solid. Select a physical unit. The default is 1 Kg.

### Custom: **Center of Mass**

Enter the center of mass coordinates with respect to the solid reference frame in the order [X Y Z]. In a uniform gravitational field, the center of mass coincides with the center of gravity. Select a physical unit. The default is [0 0 0].

### Custom: **Moments of Inertia**

Enter the mass moments of inertia of the solid element in the order [ $I_{xx}$ ,  $I_{yy}$ ,  $I_{zz}$ ]. Each moment of inertia must refer to a frame whose axes are parallel to the block reference frame axes and whose origin is coincident with the solid center of mass. The moments of inertia are the diagonal elements of the solid inertia tensor,

$$\begin{pmatrix} I_{xx} & & \\ & I_{yy} & \\ & & I_{zz} \end{pmatrix},$$

where:

- $I_{xx} = \int_V (y^2 + z^2) dm$
- $I_{yy} = \int_V (x^2 + z^2) dm$

$$\bullet \quad I_{zz} = \int_V (x^2 + y^2) dm$$

Select a physical unit. The default is [1 1 1] kg\*m^2.

**Custom: Products of Inertia**

Enter the mass products of inertia of the solid element in the order [I<sub>yz</sub>, I<sub>zx</sub>, I<sub>xy</sub>]. Each product of inertia must refer to a frame whose axes are parallel to the block reference frame axes and whose origin is coincident with the solid center of mass. The products of inertia are the off-diagonal elements of the solid inertia tensor,

$$\begin{pmatrix} & I_{xy} & I_{zx} \\ I_{xy} & & I_{yz} \\ I_{zx} & I_{yz} & \end{pmatrix},$$

where:

$$\bullet \quad I_{yz} = -\int_V yz dm$$

$$\bullet \quad I_{zx} = -\int_V zx dm$$

$$\bullet \quad I_{xy} = -\int_V xy dm$$

Select a physical unit. The default is [0 0 0] kg\*m^2.

**Graphic**

**Type**

Select a method to represent the solid in Mechanics Explorer. The default is From Geometry.

Type	Description
From Geometry	Shape specified in <b>Geometry</b> section



Type	Description
Marker	Simple icon such as Sphere, Cube, or Frame
None	No visualization

**Marker: Shape**

Geometric shape of the graphic marker. Options include Cube, Frame, and Sphere. The default setting is Sphere.

**Marker: Size**

Absolute size of the graphics marker in pixels. Changing the zoom level in the model visualization pane has no effect on the apparent marker size. The default value is 10.

**Visual Properties**

Color specification type. Options include Simple and Advanced. Select Simple to specify the solid color and opacity. Select Advanced to add lighting effects such as specular reflections and light emission.

**Simple: Color**

[R G B] color vector. This vector contains the red (R), green (G), and blue (B) contents of the solid color on a scale of 0–1. The default vector is [0.5 0.5 0.5]. A color picker provides an alternative means of defining the solid color.

**Simple: Opacity**

Degree to which the solid obscures model components positioned behind it. The opacity value can range from 0 to 1. An opacity of 0 makes the solid completely translucent, while an opacity of 1 makes the solid completely opaque. The default value is 1.0.

**Advanced: Diffuse Color**

[R G B] or [R G B A] diffuse color vector. The diffuse color is the apparent color of the solid under direct white light. The color vector contains the red (R), green (G), and blue (B) contents of the diffuse color on a scale of 0–1. It can include an optional opacity value (A), also on a scale of 0–1. The default vector is [0.5 0.5 0.5].

**Advanced: Specular Color**

[R G B] or [R G B A] specular color vector. The specular color is the color of the glossy highlights on the solid surface. The color vector contains the red (R), green (G), and blue (B) contents of the specular color on a scale of 0–1. It can include an optional opacity value (A), also on a scale of 0–1. The default vector is [0.5 0.5 0.5 1.0].

**Advanced: Ambient Color**

[R G B] or [R G B A] ambient color vector. The ambient color is the apparent color of the solid under indirect ambient light. The color vector contains the red (R), green (G), and blue (B) contents of the ambient color on a scale of 0–1. It can include an optional opacity value (A), also on a scale of 0–1. The default vector is [0.15 0.15 0.15 1.0].

**Advanced: Emissive Color**

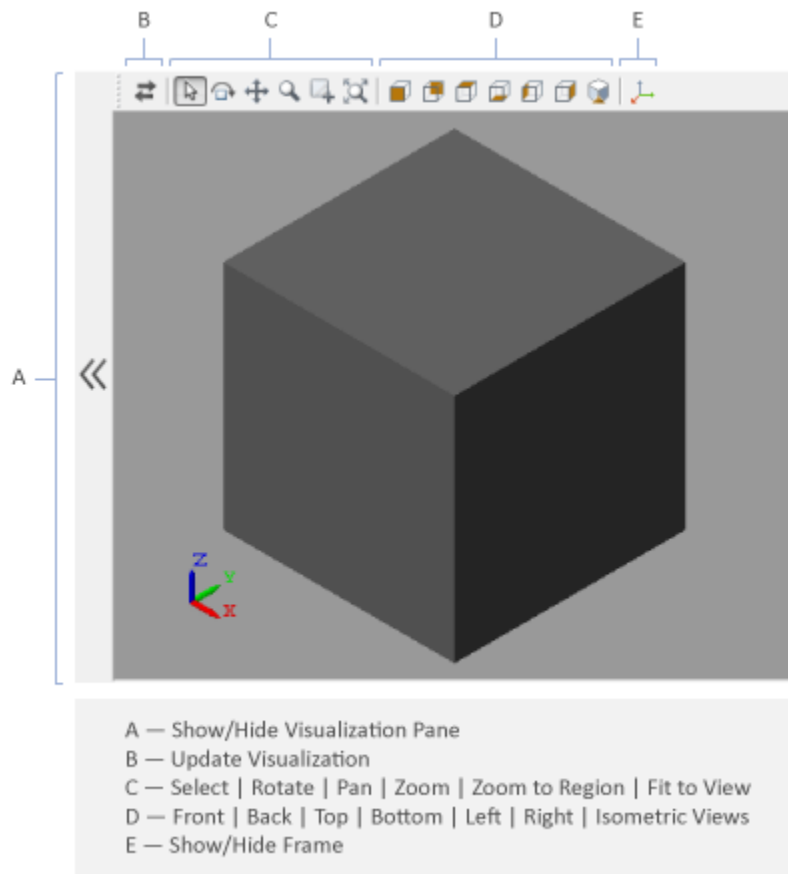
[R G B] or [R G B A] emissive color vector. The emissive color is the color of light generated by the solid. The color vector contains the red (R), green (G), and blue (B) contents of the emissive color on a scale of 0–1. It can include an optional opacity value (A), also on a scale of 0–1. The default vector is [0.0 0.0 0.0 1.0].

**Advanced: Shininess**

Sharpness of the specular highlights on the solid surface. The shininess value can range from 0 to 128. A low shininess value produces large specular highlights with a gradual falloff in intensity. A large shininess value produces small specular highlights with a sharp falloff in intensity. The default value is 75.

## **Visualization Pane**

So that you can visually check the solid you are modeling, the Solid block dialog box provides a visualization pane. This pane enables you to view the solid shape, color, and reference frame location from various 3-D perspectives. You can select a standard view point or rotate, pan, and zoom the solid view. Select the Update Visualization button to view the latest changes to the solid.



### Solid Visualization Pane

## Ports

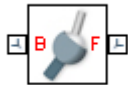
Frame port R identifies the solid reference frame.

# Spherical Joint

Joint with one spherical primitive

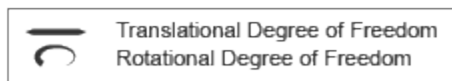
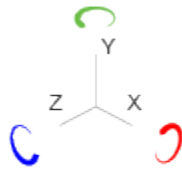
## Library

Joints



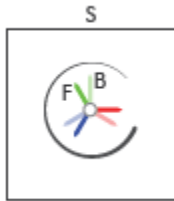
## Description

This block represents a joint with three rotational degrees of freedom. One spherical primitive provides the three rotational degrees of freedom. The base and follower frame origins remain coincident during simulation.



## Joint Degrees of Freedom

The joint block represents motion between the base and follower frames as a single time-varying transformation. The spherical primitive (S) applies this transformation, which causes the follower frame to rotate with respect to the base frame about an arbitrary 3-D axis. This joint primitive is not susceptible to gimbal lock.



### Joint Transformation

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

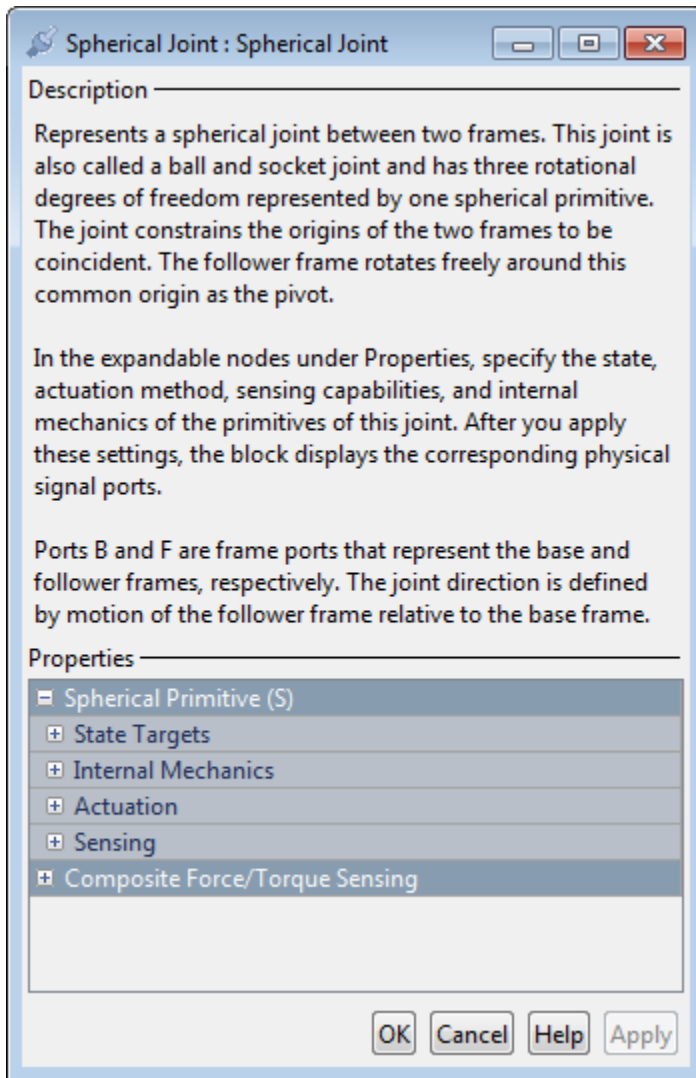
Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types: Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.



## Spherical Primitive: State Targets

Specify the spherical primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target

must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative rotation of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Select a method to specify the joint primitive state target.

Method	Description
None	Align base and follower frames.
Aligned Axes	Align two follower frame axes with two base frame axes.
Standard Axis	Specify a rotation angle about a standard axis ( $x$ , $y$ , or $z$ ).
Arbitrary Axis	Specify a rotation angle about a general $[x, y, z]$ axis.
Rotation Matrix	Specify a right-handed orthogonal rotation matrix.

### Aligned Axes

Select two pairs of base-follower frame axes.

Parameter	Description
<b>Pair 1</b>	First pair of base-follower frame axes to align.
<b>Pair 2</b>	Second pair of base-follower frame axes to align. Axis choices depend on <b>Pair 1</b> axis selections.

#### Standard Axis

Select a standard rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
<b>Axis</b>	Standard rotation axis (X, Y, or Z) resolved in the base frame.
<b>Angle</b>	Follower frame rotation angle about the rotation axis with respect to the base frame.

#### Arbitrary Axis

Select a general 3-D rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
<b>Axis</b>	General rotation axis [X Y Z] resolved in the base frame.
<b>Angle</b>	Follower frame rotation angle about the rotation axis with respect to the base frame.

#### Rotation Matrix

Specify the 3×3 transformation matrix of a proper rotation between the base and follower frames. The matrix must be orthogonal and have determinant +1. The default value is [1 0 0; 0 1 0; 0 0 1].

#### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative angular velocity of the follower frame with respect to the base frame. It



is resolved in the base frame. Selecting this option exposes priority and value fields. The priority field is identical to that used for the position state target.

### Value

Enter a three element vector with the angular velocity components of the follower frame with respect to the base frame. Select a physical unit. The default is **deg/s** (degree/second).

### Resolution Frame

From the drop-down list, select a resolution frame. This is the frame in which the state target is resolved. It identifies the axes the velocity vector components apply to. The default is **Follower**.

## Spherical Primitive: Internal Mechanics

Specify the spherical primitive internal mechanics. This includes linear spring and damping forces, accounting for energy storage and dissipation, respectively. To ignore internal mechanics, keep spring stiffness and damping coefficient values at the default value of 0.

### Equilibrium Position

Select a method to specify the spring equilibrium position. The equilibrium position is the rotation angle between base and follower port frames at which the spring torque is zero.

Method	Description
None	Align base and follower frames.
Aligned Axes	Align two follower frame axes with two base frame axes.
Standard Axis	Specify a rotation angle about a standard axis ( $x$ , $y$ , or $z$ ).
Arbitrary Axis	Specify a rotation angle about a general $[x, y, z]$ axis.
Rotation Matrix	Specify a right-handed orthogonal rotation matrix.

### Aligned Axes

Select two pairs of base-follower frame axes.

Parameter	Description
<b>Pair 1</b>	First pair of base-follower frame axes to align.
<b>Pair 2</b>	Second pair of base-follower frame axes to align. Axis choices depend on <b>Pair 1</b> axis selections.

### Standard Axis

Select a standard rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
<b>Axis</b>	Standard rotation axis (X, Y, or Z) resolved in the base frame.
<b>Angle</b>	Follower frame rotation angle about the rotation axis with respect to the base frame.

### Arbitrary Axis

Select a general 3-D rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
<b>Axis</b>	General rotation axis [X Y Z] resolved in the base frame.
<b>Angle</b>	Follower frame rotation angle about the rotation axis with respect to the base frame.

### Rotation Matrix

Specify the 3×3 transformation matrix of a proper rotation between the base and follower frames. The matrix must be orthogonal and have determinant +1. The default value is [1 0 0; 0 1 0; 0 0 1].

### Spring Stiffness

Enter the linear spring constant. This is the torque required to displace the joint primitive by a unit angle. The term linear refers to the mathematical form of the spring equation. The default is 0. Select a physical unit. The default is N\*m/deg.

### Damping Coefficient

Enter the linear damping coefficient. This is the torque required to maintain a constant joint primitive angular velocity between base and follower frames. The default is 0. Select a physical unit. The default is  $\text{N}\cdot\text{m}/(\text{deg}/\text{s})$ .

## Spherical Primitive: Actuation

Specify actuation options for the spherical joint primitive. Actuation modes include **Torque** only. Selecting a torque input adds the corresponding physical signal port to the block. Use this port to specify the actuation torque signal.

### Torque

Select a source for the actuation torque. The default setting is **None**.

Actuation Torque Setting	Description
None	Apply no actuation torque.
Provided by Input	Apply an actuation torque based on a physical signal. The signal specifies the torque acting on the follower frame with respect to the base frame. An equal and opposite torque acts on the base frame. Selecting this option exposes additional parameters.

### Torque (X), Torque (Y), Torque (Z)

Select in order to actuate the spherical joint primitive about each standard Cartesian axis (X, Y, Z) separately. The block exposes the corresponding physical signal ports. Use these ports to specify the actuation torque signals. The signals must be scalar values.

### Torque (XYZ)

Select in order to actuate the spherical joint primitive about an arbitrary axis [X Y Z]. The block exposes the corresponding physical signal port. Use this port to specify the actuation torque signal. The signal must be a 3-D vector.

### Frame

Select the frame to resolve the actuation torque signal in. The axes of this frame establish the directions of the X, Y, and Z torque components. The default setting is **Base**.

## Spherical Primitive: Sensing

Select the motion variables to sense in the spherical joint primitive. The block adds the corresponding physical signal ports. Use these ports to output the numerical values of the motion variables.

The block measures each motion variable for the follower frame with respect to the base frame. It resolves that variable in the resolution frame that you select from the **Frame** drop-down list.

Motion Variables	Description
<b>Position</b>	<p>Quaternion describing follower frame rotation with respect to base frame. The quaternion coefficients are</p> $\left[ \cos\left(\frac{\theta}{2}\right), n_x \sin\left(\frac{\theta}{2}\right), n_y \sin\left(\frac{\theta}{2}\right), n_z \sin\left(\frac{\theta}{2}\right) \right]$ <p>The measurement is the same in all measurement frames.</p>
<b>Velocity (X), Velocity (Y), Velocity (Z)</b>	Angular velocity components about X, Y, and Z axes.
<b>Velocity</b>	3-D angular velocity vector with components about X, Y, and Z axes.
<b>Acceleration (X), Acceleration (Y), Acceleration (Z)</b>	Angular acceleration components about X, Y, and Z axes.
<b>Acceleration</b>	3-D angular acceleration vector with components about X, Y, and Z axes.

### Frame

Select the frame to resolve the measurement in. The axes of this frame establish the directions of X, Y, and Z vector components. The default setting is **Base**.

## Composite Force/Torque Sensing

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

### **Direction**

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

### **Resolution Frame**

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

### **Constraint Force**

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector  $[f_{c_x}, f_{c_y}, f_{c_z}]$  acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port  $f_c$ .

### **Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque

between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector  $[tc_x, tc_y, tc_z]$  acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port  $tc$ .

### **Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector  $[ft_x, ft_y, ft_z]$ . Selecting this option causes the block to expose physical signal port  $ft$ .

### **Total Torque**

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.
- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port  $tt$ .

## **Ports**

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes

additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
t	Torque	Actuation input	Scalar if associated with an axis (e.g., tx). Three-element vector otherwise (e.g., t).
Q	Quaternion rotation	Sensing output	Four-element vector
w	Angular velocity	Sensing output	Scalar if associated with an axis (e.g., wx). Three-element vector otherwise (e.g., w).
b	Angular acceleration	Sensing output	Scalar if associated with an axis (e.g., bx). Three-element vector otherwise (e.g., b).

## See Also

6-DOF Joint | Gimbal Joint | Bushing Joint | Revolute joint

## More About

- “Motion Sensing”
- “Measurement Frames”
- “Actuating and Sensing Using Physical Signals”

# Spring and Damper Force

Force proportional to the distance and relative velocity between two frame origins

## Library

Forces and Torques



## Description

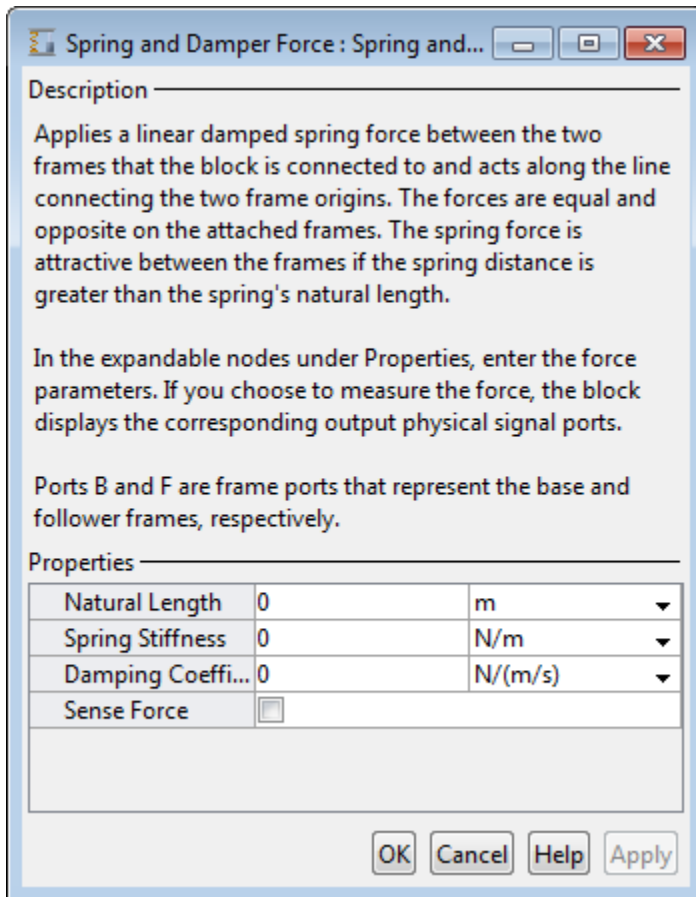
This block represents a linear spring and damper force pair acting reciprocally between base and follower frame origins. The two forces in the pair have equal magnitude but opposite directions. One force acts on the base frame origin, along the vector connecting follower to base frame origins. The other force acts on the follower frame origin, along the vector connecting base to follower frame origins.

The magnitude of the spring force component is proportional to the distance between base and follower frame origins. This distance is the length of the straight line segment connecting the two origins. The magnitude of the damper force component is proportional to the relative velocity of the follower frame origin with respect to the base frame.

## Dialog Box and Parameters

The dialog box contains a **Properties** area with three parameters that you can specify and one option that you can select.





### Natural Length

Enter the equilibrium distance between the base and follower frame origins. This is the distance at which the magnitude of the spring force is zero. The default value is 0. Select or enter a physical unit.

### Spring Stiffness

Enter the value of the linear spring constant. The value must be greater than or equal to zero. The default value is zero. Select or enter a physical unit.

### Damping Coefficient

Enter the value of the linear damping coefficient. The value must be greater than or equal to zero. The default value is zero. Select or enter a physical unit.

### **Sense Force**

Select to sense the signed magnitude of the spring and damper force acting between the two frame origins. The block exposes an additional physical signal port to output the force signal. The output signal is a scalar value. This value is positive if the force is repulsive; it is negative if the force is attractive.

### **Ports**

The block contains frame ports B and F, representing base and follower frames, respectively.

Selecting the **Sense Force** check box in the block dialog box adds a physical signal port, **fm**.

### **See Also**

External Force and Torque | Internal Force | Inverse Square Law Force

### **More About**

- “Actuating and Sensing Using Physical Signals”

# Telescoping Joint

Joint with one prismatic and one spherical joint primitive

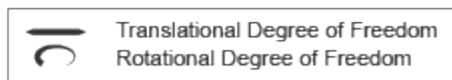
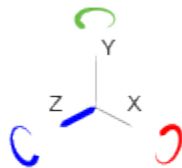
## Library

Joints



## Description

This block represents a joint with one translational and three rotational degrees of freedom. One prismatic primitive provides the translational degree of freedom. One spherical primitive provides the three rotational degrees of freedom.



## Joint Degrees of Freedom

The joint block represents motion between the base and follower frames as a sequence of time-varying transformations. Each joint primitive applies one transformation in this sequence. The transformation translates or rotates the follower frame with respect to the joint primitive base frame. For all but the first joint primitive, the base frame coincides with the follower frame of the previous joint primitive in the sequence.

At each time step during the simulation, the joint block applies the sequence of time-varying frame transformations in this order:

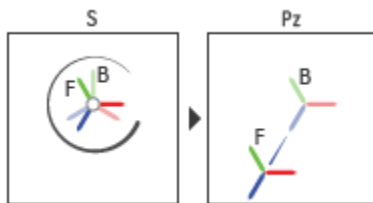
**1** Rotation:

- About an arbitrary 3-D axis resolved in the Spherical Primitive (S) base frame.

**2** Translation:

- Along the Z axis of the Z Prismatic Primitive (Pz) base frame. This frame is coincident with the Spherical Primitive (S) follower frame.

The figure shows the sequence in which the joint transformations occur at a given simulation time step. The resulting frame of each transformation serves as the base frame for the following transformation. Because 3-D rotation occurs as a single rotation about an arbitrary 3-D axis (as opposed to three separate rotations about the X, Y, Z axes), gimbal lock does not occur.



### Joint Transformation Sequence

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

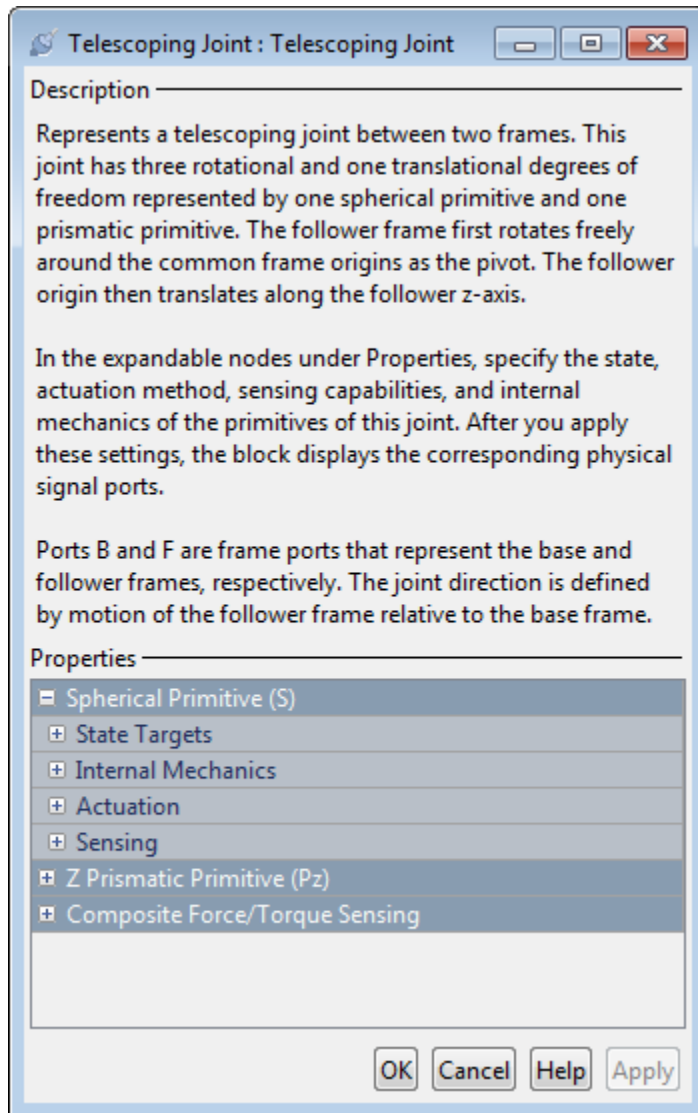
Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types: Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.



## Spherical Primitive: State Targets

Specify the spherical primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority

level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative rotation of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Select a method to specify the joint primitive state target.

Method	Description
None	Align base and follower frames.
Aligned Axes	Align two follower frame axes with two base frame axes.
Standard Axis	Specify a rotation angle about a standard axis ( $x$ , $y$ , or $z$ ).
Arbitrary Axis	Specify a rotation angle about a general $[x, y, z]$ axis.
Rotation Matrix	Specify a right-handed orthogonal rotation matrix.

### Aligned Axes

Select two pairs of base-follower frame axes.

Parameter	Description
<b>Pair 1</b>	First pair of base-follower frame axes to align.
<b>Pair 2</b>	Second pair of base-follower frame axes to align. Axis choices depend on <b>Pair 1</b> axis selections.

### Standard Axis

Select a standard rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
<b>Axis</b>	Standard rotation axis (X, Y, or Z) resolved in the base frame.
<b>Angle</b>	Follower frame rotation angle about the rotation axis with respect to the base frame.

### Arbitrary Axis

Select a general 3-D rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
<b>Axis</b>	General rotation axis [X Y Z] resolved in the base frame.
<b>Angle</b>	Follower frame rotation angle about the rotation axis with respect to the base frame.

### Rotation Matrix



Specify the 3×3 transformation matrix of a proper rotation between the base and follower frames. The matrix must be orthogonal and have determinant +1. The default value is [1 0 0; 0 1 0; 0 0 1].

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative angular velocity of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields. The priority field is identical to that used for the position state target.

#### Value

Enter a three element vector with the angular velocity components of the follower frame with respect to the base frame. Select a physical unit. The default is deg/s (degree/second).

#### Resolution Frame

From the drop-down list, select a resolution frame. This is the frame in which the state target is resolved. It identifies the axes the velocity vector components apply to. The default is Follower.

## Spherical Primitive: Internal Mechanics

Specify the spherical primitive internal mechanics. This includes linear spring and damping forces, accounting for energy storage and dissipation, respectively. To ignore internal mechanics, keep spring stiffness and damping coefficient values at the default value of 0.

### Equilibrium Position

Select a method to specify the spring equilibrium position. The equilibrium position is the rotation angle between base and follower port frames at which the spring torque is zero.

Method	Description
None	Align base and follower frames.
Aligned Axes	Align two follower frame axes with two base frame axes.
Standard Axis	Specify a rotation angle about a standard axis ( $x$ , $y$ , or $z$ ).

Method	Description
Arbitrary Axis	Specify a rotation angle about a general $[x, y, z]$ axis.
Rotation Matrix	Specify a right-handed orthogonal rotation matrix.

### Aligned Axes

Select two pairs of base-follower frame axes.

Parameter	Description
Pair 1	First pair of base-follower frame axes to align.
Pair 2	Second pair of base-follower frame axes to align. Axis choices depend on <b>Pair 1</b> axis selections.

### Standard Axis

Select a standard rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
Axis	Standard rotation axis (X, Y, or Z) resolved in the base frame.
Angle	Follower frame rotation angle about the rotation axis with respect to the base frame.

### Arbitrary Axis

Select a general 3-D rotation axis, resolved in the base frame, and specify the follower frame rotation angle.

Parameter	Description
Axis	General rotation axis $[X Y Z]$ resolved in the base frame.

Parameter	Description
Angle	Follower frame rotation angle about the rotation axis with respect to the base frame.

### Rotation Matrix

Specify the 3×3 transformation matrix of a proper rotation between the base and follower frames. The matrix must be orthogonal and have determinant +1. The default value is [1 0 0; 0 1 0; 0 0 1].

### Spring Stiffness

Enter the linear spring constant. This is the torque required to displace the joint primitive by a unit angle. The term linear refers to the mathematical form of the spring equation. The default is 0. Select a physical unit. The default is N\*m/deg.

### Damping Coefficient

Enter the linear damping coefficient. This is the torque required to maintain a constant joint primitive angular velocity between base and follower frames. The default is 0. Select a physical unit. The default is N\*m/(deg/s).

## Spherical Primitive: Actuation

Specify actuation options for the spherical joint primitive. Actuation modes include **Torque** only. Selecting a torque input adds the corresponding physical signal port to the block. Use this port to specify the actuation torque signal.

### Torque

Select a source for the actuation torque. The default setting is None.

Actuation Torque Setting	Description
None	Apply no actuation torque.
Provided by Input	Apply an actuation torque based on a physical signal. The signal specifies the torque acting on the follower frame with respect to the base frame. An equal and opposite torque acts on the base frame. Selecting this option exposes additional parameters.

**Torque (X), Torque (Y), Torque (Z)**

Select in order to actuate the spherical joint primitive about each standard Cartesian axis (X, Y, Z) separately. The block exposes the corresponding physical signal ports. Use these ports to specify the actuation torque signals. The signals must be scalar values.

**Torque (XYZ)**

Select in order to actuate the spherical joint primitive about an arbitrary axis [X Y Z]. The block exposes the corresponding physical signal port. Use this port to specify the actuation torque signal. The signal must be a 3-D vector.

**Frame**

Select the frame to resolve the actuation torque signal in. The axes of this frame establish the directions of the X, Y, and Z torque components. The default setting is Base.

**Spherical Primitive: Sensing**

Select the motion variables to sense in the spherical joint primitive. The block adds the corresponding physical signal ports. Use these ports to output the numerical values of the motion variables.

The block measures each motion variable for the follower frame with respect to the base frame. It resolves that variable in the resolution frame that you select from the **Frame** drop-down list.

Motion Variables	Description
<p><b>Position</b></p>	<p>Quaternion describing follower frame rotation with respect to base frame. The quaternion coefficients are</p> $\left[ \cos\left(\frac{\theta}{2}\right), n_x \sin\left(\frac{\theta}{2}\right), n_y \sin\left(\frac{\theta}{2}\right), n_z \sin\left(\frac{\theta}{2}\right) \right]$ <p>The measurement is the same in all measurement frames.</p>
<p><b>Velocity (X), Velocity (Y), Velocity (Z)</b></p>	<p>Angular velocity components about X, Y, and Z axes.</p>
<p><b>Velocity</b></p>	<p>3-D angular velocity vector with components about X, Y, and Z axes.</p>

Motion Variables	Description
Acceleration (X), Acceleration (Y), Acceleration (Z)	Angular acceleration components about X, Y, and Z axes.
Acceleration	3-D angular acceleration vector with components about X, Y, and Z axes.

### Frame

Select the frame to resolve the measurement in. The axes of this frame establish the directions of X, Y, and Z vector components. The default setting is **Base**.

## Prismatic Primitive: State Targets

Specify the prismatic primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative position, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative velocity, measured along the joint primitive axis, of the follower frame origin with respect to the base frame origin. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely

Priority Level	Description
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is m for position and m/s for velocity.

## Prismatic Primitive: Internal Mechanics

Specify the prismatic primitive internal mechanics. Internal mechanics include linear spring forces, accounting for energy storage, and damping forces, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

### Equilibrium Position

Enter the spring equilibrium position. This is the distance between base and follower frame origins at which the spring force is zero. The default value is 0. Select or enter a physical unit. The default is m.

### Spring Stiffness

Enter the linear spring constant. This is the force required to displace the joint primitive by a unit distance. The default is 0. Select or enter a physical unit. The default is N/m.

### Damping Coefficient

Enter the linear damping coefficient. This is the force required to maintain a constant joint primitive velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is N/(m/s).

## Prismatic Primitive: Actuation

Specify actuation options for the prismatic joint primitive. Actuation modes include **Force** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Actuation signals are resolved in the base frame.

## Force

Select an actuation force setting. The default setting is **None**.

Actuation Force Setting	Description
None	No actuation force.
Provided by Input	Actuation force from physical signal input. The signal provides the force acting on the follower frame with respect to the base frame along the joint primitive axis. An equal and opposite force acts on the base frame.
Automatically computed	Actuation force from automatic calculation. SimMechanics computes and applies the actuation force based on model dynamics.

## Motion

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

## Prismatic Primitive: Sensing

Select the variables to sense in the prismatic joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved

in the base frame. You can use the measurement signals for analysis or as input in a control system.

**Position**

Select this option to sense the relative position of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Velocity**

Select this option to sense the relative velocity of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Acceleration**

Select this option to sense the relative acceleration of the follower frame origin with respect to the base frame origin along the joint primitive axis.

**Actuator Force**

Select this option to sense the actuation force acting on the follower frame with respect to the base frame along the joint primitive axis.

**Composite Force/Torque Sensing**

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

**Direction**

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

**Resolution Frame**



You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

### **Constraint Force**

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector  $[fc_x, fc_y, fc_z]$  acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

### **Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector  $[tc_x, tc_y, tc_z]$  acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

### **Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force

vector  $[ft_x, ft_y, ft_z]$ . Selecting this option causes the block to expose physical signal port  $ft$ .

### Total Torque

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.
- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port  $tt$ .

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
f	Force	Actuation input	Scalar if associated with an axis (e.g. fx). Three-element vector otherwise (e.g. f).
t	Torque	Actuation input	Scalar if associated with an axis (e.g. tx). Three-element vector otherwise (e.g. t).

Port Label	Description	Type	Input/Output
p	Position	Sensing output	Scalar if associated with an axis (e.g. px). Three-element vector otherwise (e.g. p).
v	Velocity	Sensing output	Scalar if associated with an axis (e.g. vx). Three-element vector otherwise (e.g. v).
a	Acceleration	Sensing output	Scalar if associated with an axis (e.g. ax). Three-element vector otherwise (e.g. a).
Q	Quaternion rotation	Sensing output	Four-element vector
w	Angular velocity	Sensing output	Scalar if associated with an axis (e.g. wx). Three-element vector otherwise (e.g. w).
b	Angular acceleration	Sensing output	Scalar if associated with an axis (e.g. bx). Three-element vector otherwise (e.g. b).

## See Also

For related blocks, see Prismatic Joint, Spherical Joint

For more information, see:

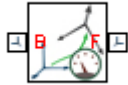
- “Motion Sensing”
- “Measurement Frames”
- “Actuating and Sensing Using Physical Signals”

# Transform Sensor

Sensor that measures the spatial relationship between two frames

## Library

Frames and Transforms



## Description

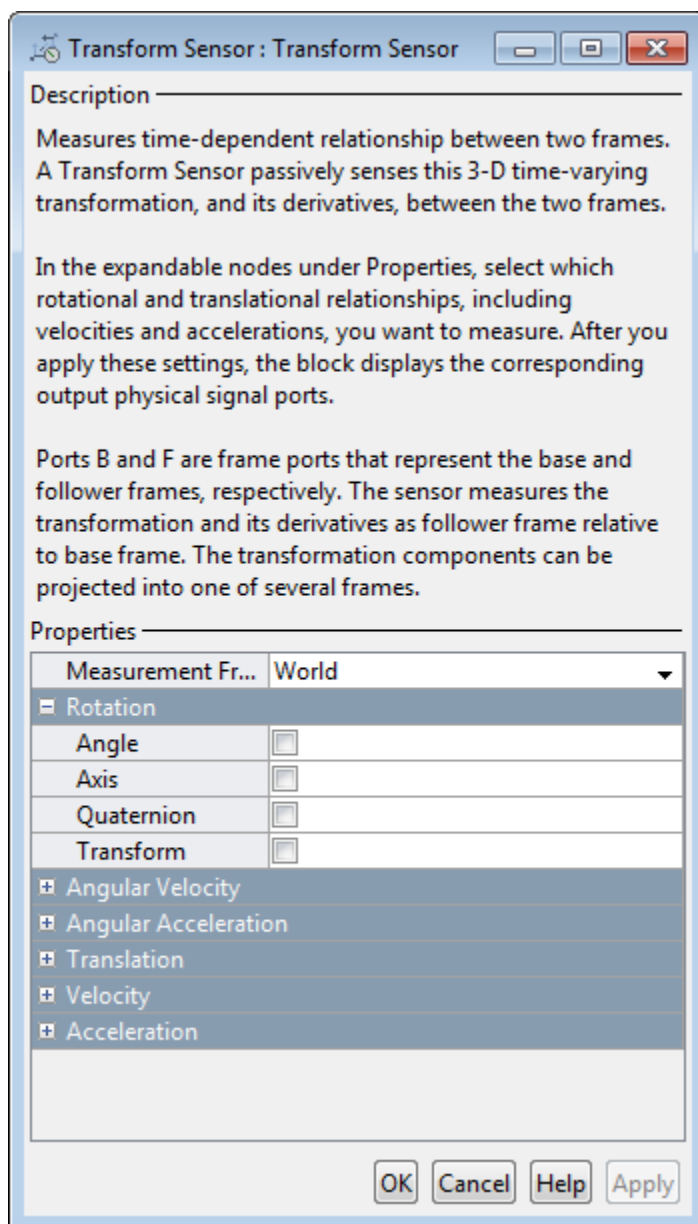
This block represents a sensor that measures the spatial relationship between two frames. Parameters that this sensor measures include rotational and translational position, velocity, and acceleration. The sensor can measure these parameters between any two frames in a model. This block provides the broadest motion sensing capability in SimMechanics.

Each measurement provides the value of a parameter for the follower frame with respect to the base frame, resolved in the **Measurement Frame** that you choose. Measurement frames include World as well as rotating and non-rotating base and follower frames.

To output a parameter, the block provides a set of optional physical signal ports. Ports remain hidden until you select the corresponding parameters in the dialog box. Each port outputs a parameter as a time-varying physical signal. By default, measurements are in SI units. If connecting to Simulink® blocks, you can use the PS-Simulink Converter block to select a different physical unit.

## Dialog Box and Parameters

The dialog box contains a **Properties** area with expandable angular and linear measurement sections:



### Measurement Frame

Select a frame in which to resolve the selected spatial measurements. The choice of measurement frame affects the expression of a vector quantity in terms of its X, Y, and Z components. Some quantities, such as **Angle**, are not affected by the choice of measurement frame. For more information, see “Measurement Frames”. The default is **World**.

## **Rotation**

Select the rotation parameters to sense. These parameters encode the rotation operation required to bring the base frame into coincidence with the follower frame. Rotation observes the right-hand rule: with the rotation axis pointing out of the screen, counterclockwise motion defines positive rotation, while clockwise motion defines negative rotation.

Non-vector quantities require no measurement frame for resolution; these quantities are unaffected by measurement frame choice. Vector quantities, such as **Axis**, are *always* resolved in either base or follower measurement frames; the **World** measurement frame does not apply.

### **Angle**

3-D rotation angle of the follower frame with respect to the base frame. Selecting **Angle** exposes physical signal port **q**.

### **Axis**

Vector components of the normalized rotation axis. The output is a three-element vector with the X, Y, and Z axis components resolved in the measurement frame. Selecting **Axis** exposes physical signal port **axs**.

### **Quaternion**

Unit quaternion that describes the pure rotation of the follower frame with respect to the base frame. The output is a four-element vector with the scalar (*S*) and vector ( $V_x, V_y, V_z$ ) quaternion coefficients. The vector provides the coefficients in the order [*S*  $V_x$   $V_y$   $V_z$ ]. Selecting **Quaternion** exposes physical signal port **Q**.

### **Transform**

Transform matrix that describes the pure rotation of the follower frame with respect to the base frame. The output is a nine-element, 3×3 matrix. Selecting **Transform** exposes physical signal port **R**.

## Angular Velocity

Select the angular velocity parameters to sense. The parameters encode the angular velocity of the follower frame with respect to the base frame, resolved in the measurement frame. Rotation observes the right-hand rule: with the rotation axis pointing out of the screen, counterclockwise motion defines positive rotation, while clockwise motion defines negative rotation.

### Omega X/Omega Y/Omega Z

Relative angular velocities about the X, Y, and Z axes of the base frame. Selecting **Omega X**, **Omega Y**, and **Omega Z** exposes physical signal ports **wx**, **wy**, and **wz**.

### Quaternion

Unit quaternion that describes the angular velocity of the follower frame with respect to the base frame. The output is a four-element vector with the scalar ( $S$ ) and vector ( $V_x, V_y, V_z$ ) quaternion coefficients. The vector provides the coefficients in the order [ $S V_x V_y V_z$ ]. Selecting **Quaternion** exposes physical signal port **Qd**.

### Transform

Transform matrix that describes the angular velocity of the follower frame with respect to the base frame. The output is a nine-element,  $3 \times 3$  matrix. Selecting **Transform** exposes physical signal port **Rd**.

## Angular Acceleration

Select the angular acceleration parameters to sense. The parameters encode the angular acceleration of the follower frame with respect to the base frame, resolved in the measurement frame. Rotation observes the right-hand rule: with the rotation axis pointing out of the screen, counterclockwise motion defines positive rotation, while clockwise motion defines negative rotation.

### Alpha X/Alpha Y/Alpha Z

Relative angular accelerations about the X, Y, and Z axes of the base frame. Selecting **Alpha X**, **Alpha Y**, and **Alpha Z** exposes physical signal ports **bx**, **by**, and **bz**.

### Quaternion

Unit quaternion that describes the angular acceleration of the follower frame with respect to the base frame. The output is a four-element vector with the scalar ( $S$ ) and vector ( $V_x, V_y, V_z$ ) quaternion coefficients. The vector provides the coefficients in the order [ $S V_x V_y V_z$ ]. Selecting **Quaternion** exposes physical signal port **Qdd**.

### Transform

Transform matrix that describes the angular acceleration of the follower frame with respect to the base frame. The output is a nine-element, 3×3 matrix. Selecting **Transform** exposes physical signal port **Rdd**.

### Translation

Select the translation parameters to sense. The parameters encode the translation of the follower frame with respect to the base frame, resolved in the measurement frame.

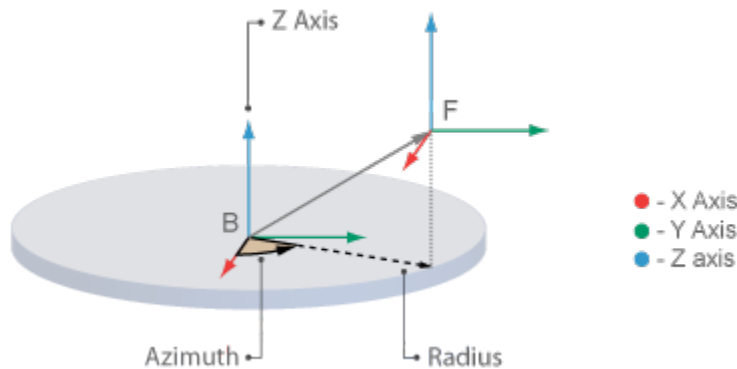
#### X/Y/Z

Offset vector from the base frame origin to the follower frame origin along the X, Y, and Z axes. Selecting **X**, **Y**, and **Z** exposes physical signal ports **x**, **y**, and **z**.

#### Radius

Standard radius coordinate found in cylindrical coordinate systems. This radius is the shortest distance from the base frame Z axis to the follower frame origin. The value of the radius is always greater than or equal to zero. Selecting **Radius** exposes physical signal port **rad**.

The figure shows the cylindrical translation parameters **Z**, **Radius**, and **Azimuth**.



#### Azimuth

Standard azimuth coordinate found in cylindrical and spherical coordinate systems. The azimuth is the angle from the base frame +X axis to the projection of the ray

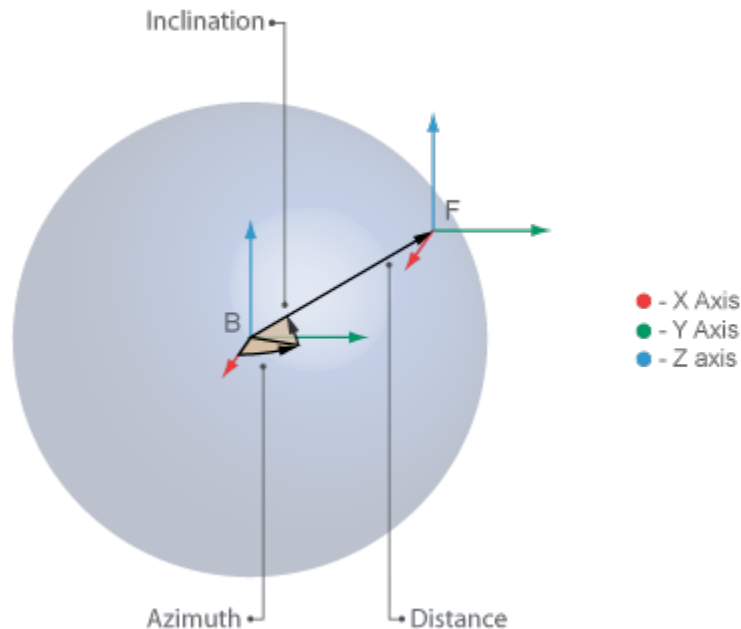


connecting base to follower frame origins onto the base frame XY plane. The angle measurement observes the right-hand rule. The azimuth falls in the range  $[-180^\circ, +180^\circ]$ . If base and follower frame origins coincide with each other, the azimuth is undefined. Selecting **Azimuth** exposes sensing port **azm**.

### Distance

Standard radius found in spherical coordinate systems. This is the distance from the origin of the base frame to the origin of the follower frame. This distance is always equal to or greater than zero. Selecting **Distance** exposes sensing port **dst**.

The figure shows the spherical translation parameters **Azimuth**, **Distance**, and **Inclination**.



### Inclination

Standard inclination found in spherical coordinate systems. The inclination is the angle of the ray connecting base to follower frame origins with respect to the projection of this ray onto the base frame XY plane. The angle measurement observes the right-hand rule. The inclination falls in the range  $[-90^\circ, +90^\circ]$ . If base

and follower frame origins coincide with each other, the inclination is undefined. Selecting **Inclination** exposes sensing port **inc**.

## Velocity

Select the linear velocity parameters to sense. The parameters encode the linear velocity of the follower frame with respect to the base frame, resolved in the measurement frame. Differentiation of a translation parameter occurs in measurement coordinates, after that parameter is resolved in the chosen measurement frame.

### X/Y/Z

Relative linear velocities along the X, Y, and Z axes. Selecting **X**, **Y**, and **Z** exposes physical signal ports **vx**, **vy**, and **vz**.

### Radius

Time rate of change of the **Radius** coordinate defined under **Translation**. Selecting **Radius** exposes physical signal port **vrad**.

### Azimuth

Time rate of change of the **Azimuth** coordinate defined under **Translation**. Selecting **Azimuth** exposes physical signal port **vazm**.

### Distance

Time rate of change of the **Distance** coordinate defined under **Translation**. Selecting **Distance** exposes physical signal port **vdst**.

### Inclination

Time rate of change of the **Inclination** coordinate defined under **Translation**. Selecting **Inclination** exposes physical signal port **vinc**.

## Acceleration

Select the linear acceleration parameters to sense. The parameters encode the linear acceleration of the follower frame with respect to the base frame, resolved in the measurement frame. Differentiation of a translation parameter occurs in measurement coordinates, after that parameter is resolved in the chosen measurement frame.

### X/Y/Z

Relative linear accelerations along the X, Y, and Z axes. Selecting **X**, **Y**, and **Z** exposes physical signal ports **ax**, **ay**, and **az**.

**Radius**

Second time-derivative of the **Radius** coordinate defined under **Translation**.  
Selecting **Radius** exposes physical signal port **arad**.

**Azimuth**

Second time-derivative of the **Azimuth** coordinate defined under **Translation**.  
Selecting **Azimuth** exposes physical signal port **aazm**.

**Distance**

Second time-derivative of the **Distance** coordinate defined under **Translation**.  
Selecting **Distance** exposes physical signal port **adst**.

**Inclination**

Second time-derivative of the **Inclination** coordinate defined under **Translation**.  
Selecting **Inclination** exposes physical signal port **ainc**.

**Ports**

The block contains frame ports B and F, representing base and follower frames, respectively.

**See Also**

Rigid Transform

**More About**

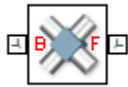
- “Working with Frames”
- “Motion Sensing”
- “Representing Frames”

# Universal Joint

Joint with two revolute primitives

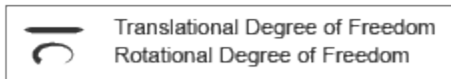
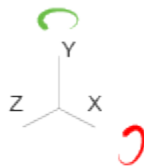
## Library

Joints



## Description

This block represents a joint with two rotational degrees of freedom. Two revolute primitives provide the two rotational degrees of freedom. The base and follower frame origins remain coincident during simulation.



## Joint Degrees of Freedom

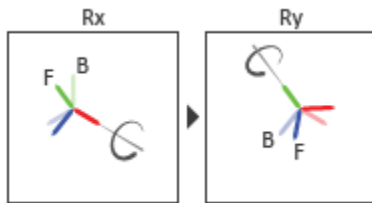
The joint block represents motion between the base and follower frames as a sequence of time-varying transformations. Each joint primitive applies one transformation in this sequence. The transformation translates or rotates the follower frame with respect to the joint primitive base frame. For all but the first joint primitive, the base frame coincides with the follower frame of the previous joint primitive in the sequence.

At each time step during the simulation, the joint block applies the sequence of time-varying frame transformations in this order:

## 1 Rotation:

- a About the X axis of the X Revolute Primitive (Rx) base frame.
- b About the Y axis of the Y Revolute Primitive (Ry) base frame. This frame is coincident with the X Revolute Primitive (Rx) follower frame.

The figure shows the sequence in which the joint transformations occur at a given simulation time step. The resulting frame of each transformation serves as the base frame for the following transformation.



### Joint Transformation Sequence

A set of optional state targets guide assembly for each joint primitive. Targets include position and velocity. A priority level sets the relative importance of the state targets. If two targets are incompatible, the priority level determines which of the targets to satisfy.

Internal mechanics parameters account for energy storage and dissipation at each joint primitive. Springs act as energy storage elements, resisting any attempt to displace the joint primitive from its equilibrium position. Joint dampers act as energy dissipation elements. Springs and dampers are strictly linear.

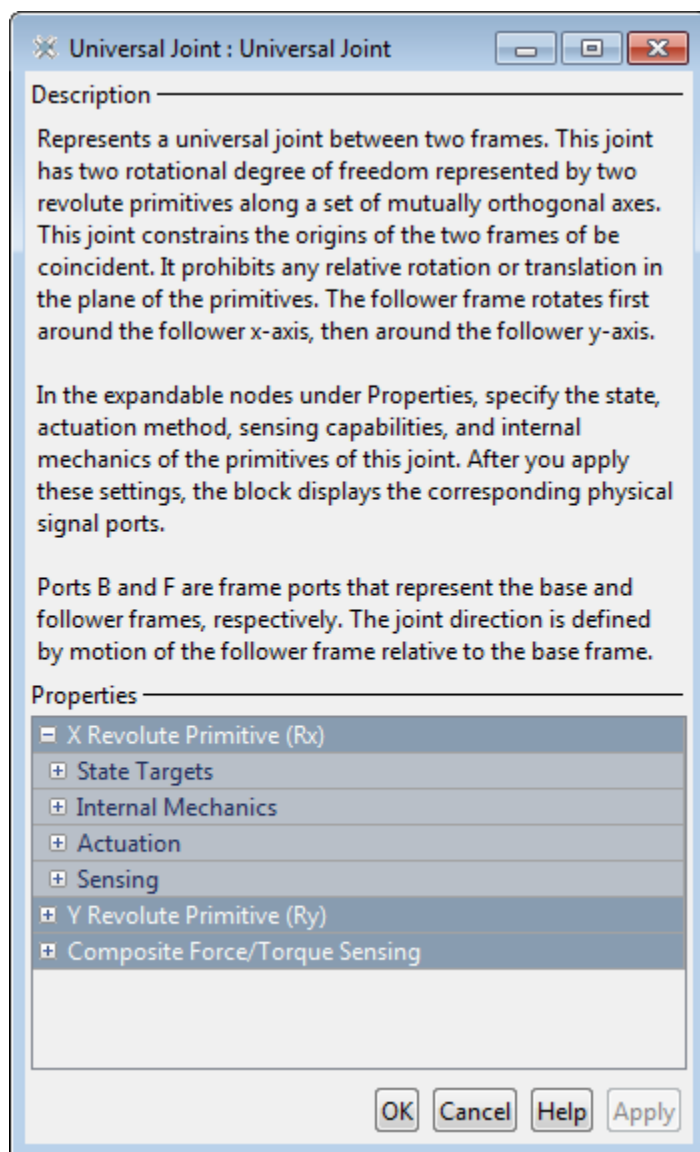
Each joint primitive has a set of optional actuation and sensing ports. Actuation ports accept physical signal inputs that drive the joint primitives. These inputs can be forces and torques or a desired joint trajectory. Sensing ports provide physical signal outputs that measure joint primitive motion as well as actuation forces and torques. Actuation modes and sensing types vary with joint primitive.

## Dialog Box and Parameters

Expandable sections provide parameters and options for the different joint primitives. These primitives are the basic elements of a joint block. They can be of three types:

Revolute, Prismatic, or Spherical. Joint blocks can have all, some, or none of these joint primitives. For example, the Weld joint block has none.

The expandable sections are hierarchical. The top level of an expandable section identifies joint primitive type and axis, e.g., **X Prismatic Primitive (Px)**. Within a joint primitive section are four parameter groups. These contain parameters and options for a joint primitive's initial state, internal mechanics, actuation, and sensing.



## Revolute Primitive: State Targets

Specify the revolute primitive state targets and their priority levels. A state target is the desired value for one of the joint state parameters—position and velocity. The priority level is the relative importance of a state target. It determines how precisely the target must be met. Use the Model Report tool in Mechanics Explorer to check the assembly status for each joint state target.

### Specify Position Target

Select this option to specify the desired joint primitive position at time zero. This is the relative rotation angle, measured about the joint primitive axis, of the follower frame with respect to the base frame. The specified target is resolved in the base frame. Selecting this option exposes priority and value fields.

### Specify Velocity Target

Select this option to specify the desired joint primitive velocity at time zero. This is the relative angular velocity, measured about the joint primitive axis, of the follower frame with respect to the base frame. It is resolved in the base frame. Selecting this option exposes priority and value fields.

### Priority

Select state target priority. This is the importance level assigned to the state target. If all state targets cannot be simultaneously satisfied, the priority level determines which targets to satisfy first and how closely to satisfy them. This option applies to both position and velocity state targets.

Priority Level	Description
High (desired)	Satisfy state target precisely
Low (approximate)	Satisfy state target approximately

---

**Note:** During assembly, high-priority targets behave as exact guides. Low-priority targets behave as rough guides.

---

### Value

Enter the state target numerical value. The default is 0. Select or enter a physical unit. The default is **deg** for position and **deg/s** for velocity.



## Revolute Primitive: Internal Mechanics

Specify the revolute primitive internal mechanics. Internal mechanics include linear spring torques, accounting for energy storage, and linear damping torques, accounting for energy dissipation. You can ignore internal mechanics by keeping spring stiffness and damping coefficient values at 0.

### Equilibrium Position

Enter the spring equilibrium position. This is the rotation angle between base and follower frames at which the spring torque is zero. The default value is 0. Select or enter a physical unit. The default is **deg**.

### Spring Stiffness

Enter the linear spring constant. This is the torque required to rotate the joint primitive by a unit angle. The default is 0. Select or enter a physical unit. The default is **N\*m/deg**.

### Damping Coefficient

Enter the linear damping coefficient. This is the torque required to maintain a constant joint primitive angular velocity between base and follower frames. The default is 0. Select or enter a physical unit. The default is **N\*m/(deg/s)**.

## Revolute Primitive: Actuation

Specify actuation options for the revolute joint primitive. Actuation modes include **Torque** and **Motion**. Selecting **Provided by Input** from the drop-down list for an actuation mode adds the corresponding physical signal port to the block. Use this port to specify the input signal. Input signals are resolved in the base frame.

### Torque

Select an actuation torque setting. The default setting is **None**.

Actuation Torque Setting	Description
None	No actuation torque.
Provided by Input	Actuation torque from physical signal input. The signal provides the torque acting on the follower frame with respect to the base frame about the joint primitive axis. An equal and opposite torque acts on the base frame.

Actuation Torque Setting	Description
Automatically computed	Actuation torque from automatic calculation. SimMechanics computes and applies the actuation torque based on model dynamics.

### Motion

Select an actuation motion setting. The default setting is **Automatically Computed**.

Actuation Motion Setting	Description
Provided by Input	Joint primitive motion from physical signal input. The signal provides the desired trajectory of the follower frame with respect to the base frame along the joint primitive axis.
Automatically computed	Joint primitive motion from automatic calculation. SimMechanics computes and applies the joint primitive motion based on model dynamics.

### Revolute Primitive: Sensing

Select the variables to sense in the revolute joint primitive. Selecting a variable exposes a physical signal port that outputs the measured quantity as a function of time. Each quantity is measured for the follower frame with respect to the base frame. It is resolved in the base frame. You can use the measurement signals for analysis or as input in a control system.

#### Position

Select this option to sense the relative rotation angle of the follower frame with respect to the base frame about the joint primitive axis.

#### Velocity

Select this option to sense the relative angular velocity of the follower frame with respect to the base frame about the joint primitive axis.

#### Acceleration

Select this option to sense the relative angular acceleration of the follower frame with respect to the base frame about the joint primitive axis.

### **Actuator Torque**

Select this option to sense the actuation torque acting on the follower frame with respect to the base frame about the joint primitive axis.

## **Composite Force/Torque Sensing**

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

### **Direction**

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

### **Resolution Frame**

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

### **Constraint Force**

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint

force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector [ $f_{c_x}$ ,  $f_{c_y}$ ,  $f_{c_z}$ ] acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

### **Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector [ $tc_x$ ,  $tc_y$ ,  $tc_z$ ] acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.

### **Total Force**

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector [ $ft_x$ ,  $ft_y$ ,  $ft_z$ ]. Selecting this option causes the block to expose physical signal port *ft*.

### **Total Torque**

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.
- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port  $tt$ .

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively. Selecting actuation or sensing options from the dialog box exposes additional physical signal ports. Use the ports to input an actuation signal or to output the chosen sensing parameter.

A unique label identifies the actuation or sensing component associated with a port. This label can contain one or two letters. The first letter identifies the actuation or sensing parameter, applied to or measured from the follower frame. The second letter identifies the axis for that parameter, resolved in the base frame. This letter can be x, y, or z.

The table describes the first letters in the port labels for this block.

Port Label	Description	Type	Input/Output
t	Torque	Actuation input	Scalar
q	Rotation angle	Sensing output	Scalar
w	Angular velocity	Sensing output	Scalar
b	Angular acceleration	Sensing output	Scalar

## See Also

Revolute Joint | Gimbal Joint

## More About

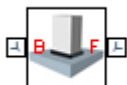
- “Actuating and Sensing Using Physical Signals”
- “Motion Sensing”
- “Rotational Measurements”

## Weld Joint

Joint with zero primitives

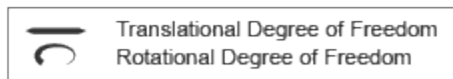
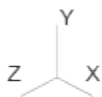
## Library

Joints



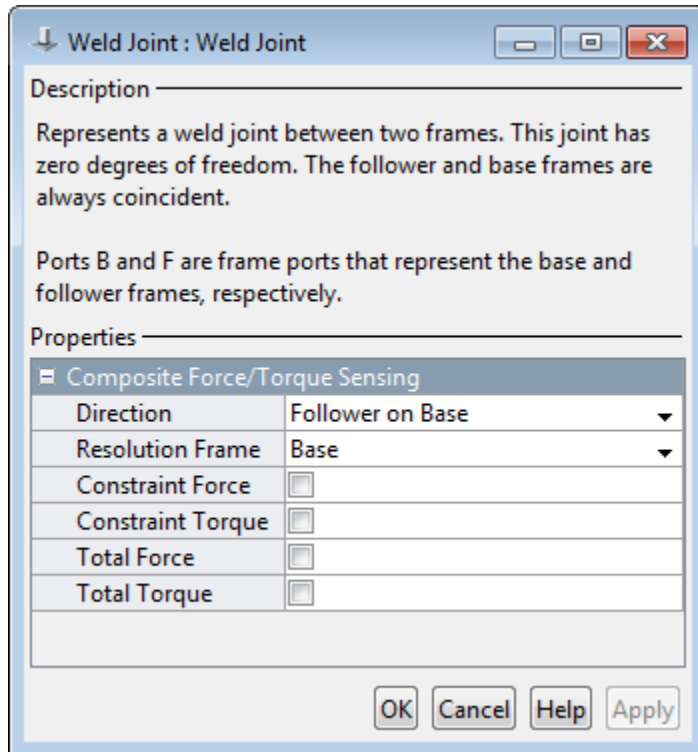
## Description

This block represents a joint with zero degrees of freedom. It contains no joint primitives. Base and follower frames, each connected to a separate rigid body, are coincident for all time. The block dialog box provides sensing options for constraint and total forces and torques.



## Joint Degrees of Freedom

## Dialog Box and Parameters



### Composite Force/Torque Sensing

Select the composite, or joint-wide, forces and torques to sense. These are forces and torques that act not at individual joint primitives but at the whole joint. Options include constraint and total forces and torques.

During simulation, the block computes the selected composite forces and torques acting between the base and follower port frames. It outputs these variables using physical signal output ports. Check the port labels to identify the output variables at different ports.

#### Direction

Forces and torques acting at joints do so in pairs. Newton's third law of motion requires that every action be accompanied by an equal and opposite reaction. If the base frame of a joint exerts a force or torque on the follower frame, then the follower frame must exert an equal and opposite force or torque on the base frame.

Select whether to sense the composite forces and torques exerted by the base frame on the follower frame or vice versa. The force and torque vector components are positive if they point along the positive X, Y, and Z axes of the selected resolution frame.

### **Resolution Frame**

You can resolve a vector quantity into Cartesian components in different frames. If the resolution frames have different orientations, then the measured components are themselves different—even though the vector quantity remains the same.

Select the frame in which to resolve the sensed force and torque variables. Possible resolution frames include **Base** and **Follower**. The block outputs the Cartesian components of the sensed force and torque vectors as observed in this frame.

### **Constraint Force**

Joint blocks with fewer than three translational degrees of freedom forbid motion along one or more axes. For example, the Gimbal Joint block forbids translation along all axes. To prevent translation along an axis, a joint block applies a constraint force between its base and follower port frames. Constraint forces are orthogonal to joint translation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint force vector [ $f_{c_x}$ ,  $f_{c_y}$ ,  $f_{c_z}$ ] acting at the joint. Only constraint force components that are orthogonal to the joint translational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *fc*.

### **Constraint Torque**

Joint blocks with fewer than three rotational degrees of freedom forbid motion about one or more axes. For example, the Cartesian Joint block forbids rotation about all axes. To prevent rotation about an axis, a joint block applies a constraint torque between its base and follower port frames. Constraint torques are orthogonal to joint rotation axes and therefore do no work.

Select the check box to compute and output the 3-D constraint torque vector [ $tc_x$ ,  $tc_y$ ,  $tc_z$ ] acting at the joint. Only constraint torque components that are orthogonal to the joint rotational degrees of freedom have nonzero values. Selecting this option causes the block to expose physical signal port *tc*.



### Total Force

A joint block generally applies various forces between its port frames:

- Actuation forces that drive prismatic joint primitives.
- Internal spring and damper forces that resist motion at prismatic joint primitives.
- Constraint forces that forbid motion in directions orthogonal to prismatic joint primitives.

The net sum of the different force components equals the total force acting between the joint port frames. Select the check box to compute and output the 3-D total force vector  $[ft_x, ft_y, ft_z]$ . Selecting this option causes the block to expose physical signal port  $ft$ .

### Total Torque

A joint block generally applies various torques between its port frames:

- Actuation torques that drive revolute or spherical joint primitives.
- Internal spring and damper torques that resist motion at revolute or spherical joint primitives.
- Constraint torques that forbid motion in directions orthogonal to the revolute or spherical joint primitive axes.

The net sum of the different torque components equals the total torque acting at a joint. Select the check box to compute and output the 3-D total torque vector  $[tt_x, tt_y, tt_z]$ . Selecting this option causes the block to expose physical signal port  $tt$ .

## Ports

The block contains frame ports B and F, representing base and follower frames, respectively.

### See Also

Rigid Transform

## World Frame

Inertial reference frame

## Library

Frames and Transforms

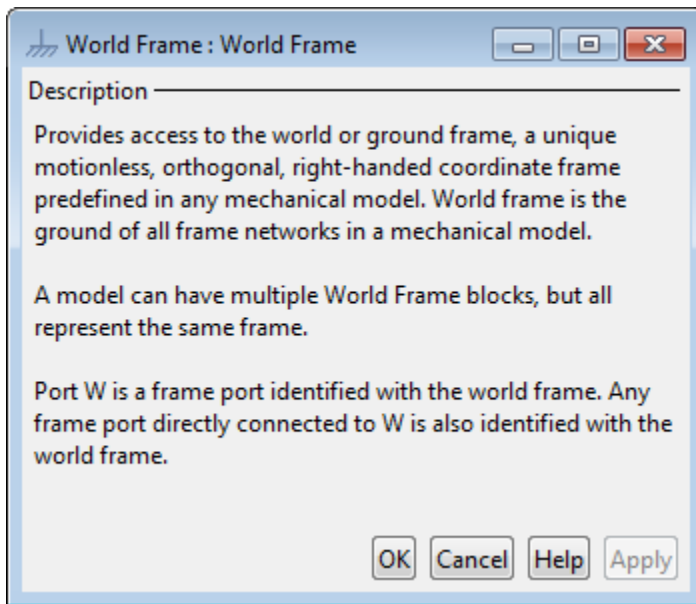


## Description

This block represents the global reference frame in a model. This frame is inertial and at absolute rest. Rigidly connecting a frame to the World frame makes that frame inertial. Frame axes are orthogonal and arranged according to the right-hand rule.

In a frame network, the World frame is the ultimate reference frame. Directly or indirectly, all other frames are defined with respect to the World frame. If multiple World Frame blocks connect to the same frame network, those blocks identify the same frame. If no World Frame block connects to a frame network, a copy of an existing frame, frozen in its initial position and orientation, serves as the World frame.

## Dialog Box and Parameters



## Ports

This block contains frame port W, representing the World frame.

## See Also

Reference Frame | Rigid Transform

## More About

- “World and Reference Frames”
- “Representing Frames”



# Configuration Parameters

---

- “SimMechanics Pane: General” on page 2-2
- “SimMechanics Pane: Diagnostics” on page 2-3
- “SimMechanics Pane: Explorer” on page 2-12

# SimMechanics Pane: General

The SimMechanics Second Generation (2G) configuration parameters are arranged into the following sections :

### Diagnostics

This section contains configurable diagnostic messages. The messages can be configured to be ignored or to be reported as warnings or errors. Errors will prevent simulation while warnings will allow simulation to proceed. The Mechanics Explorer (if selected) will be opened and visualization shown in all cases.

### Explorer

This section contains parameters that configure the Mechanics Explorer.

## SimMechanics Pane Overview

Configure the mechanical settings for an entire SimMechanics model.

### Configuration

- This pane appears only if your model contains at least one block from the Simscape product or a product based on the Simscape product, such as the SimMechanics product.
- The settings in this pane are saved only if your model contains at least one SimMechanics block.

## SimMechanics Pane: Diagnostics

Evaluation	
Invalid visual properties:	warning ▾
Repeated vertices in a cross-section:	warning ▾
Topology	
Unconnected frame port:	warning ▾
Redundant block:	warning ▾
Conflicting reference frames:	warning ▾
Rigidly constrained block:	error ▾
Assembly	
Unsatisfied high priority state targets:	warning ▾
Overspecified targets in kinematic loops:	error ▾

### In this section...

“Invalid visual properties” on page 2-4

“Repeated vertices in a cross-section” on page 2-5

“Unconnected frame port” on page 2-6

“Redundant block” on page 2-7

“Conflicting reference frames” on page 2-8

“Rigidly constrained block” on page 2-9

“Unsatisfied high priority state targets” on page 2-10

“Overspecified targets in kinematic loops” on page 2-11

### Invalid visual properties

Select the diagnostic action to take if the application detects an improperly specified color vector.

#### Settings

**Default:** warning

none

The application does not check for this situation.

warning

When the application detects this situation, it displays a warning.

error

When the application detects this situation, it terminates the simulation and displays an error message.

#### Command-Line Information

**Parameter:** SimMechanicsInvalidVisualProperty

**Type:** string

**Value:** none | warning | error

**Default:** warning



## Repeated vertices in a cross-section

Select the diagnostic action to take if the application detects repeated vertices in a cross-section.

### Settings

**Default:** warning

none

The application does not check for this situation.

warning

When the application detects this situation, it displays a warning.

error

When the application detects this situation, it terminates the simulation and displays an error message.

### Command-Line Information

**Parameter:** SimMechanicsCrossSectionNullEdge

**Type:** string

**Value:** none | warning | error

**Default:** warning

### Unconnected frame port

Select the diagnostic action to take if the application detects an unconnected frame port.

#### Settings

**Default:** Warning

none

The application does not check for this situation.

warning

When the application detects this situation, it displays a warning.

error

When the application detects this situation, it terminates the simulation and displays an error message.

#### Command-Line Information

**Parameter:** SimMechanicsUnconnectedFramePorts

**Type:** string

**Value:** none | warning | error

**Default:** warning

## Redundant block

Select the diagnostic action to take if the application detects a redundant block in the model.

### Settings

**Default:** warning

none

The application does not check for this situation.

warning

When the application detects this situation, it displays a warning.

error

When the application detects this situation, it terminates the simulation and displays an error message.

### Command-Line Information

**Parameter:** SimMechanicsRedundantBlock

**Type:** string

**Value:** none | warning | error

**Default:** warning

### Conflicting reference frames

Select the diagnostic action to take if the application detects conflicting reference frames in the model.

#### Settings

**Default:** warning

none

The application does not check for this situation.

warning

When the application detects this situation, it displays a warning.

error

When the application detects this situation, it terminates the simulation and displays an error message.

#### Command-Line Information

**Parameter:** SimMechanicsConflictingReferenceFrames

**Type:** string

**Value:** none | warning | error

**Default:** warning

## Rigidly constrained block

Select the diagnostic action to take if the application detects a rigidly constrained block in the model.

### Settings

**Default:** warning

none

The application does not check for this situation.

warning

When the application detects this situation, it displays a warning.

error

When the application detects this situation, it terminates the simulation and displays an error message.

### Command-Line Information

**Parameter:** SimMechanicsRigidlyBoundBlock

**Type:** string

**Value:** none | warning | error

**Default:** error

### Unsatisfied high priority state targets

Select the diagnostic action to take if the application detects targets with unsatisfied desired states in the model.

#### Settings

**Default:** warning

none

The application does not check for this situation.

warning

When the application detects this situation, it displays a warning.

error

When the application detects this situation, it terminates the simulation and displays an error message.

#### Command-Line Information

**Parameter:** SimMechanicsUnsatisfiedHighPriorityTargets

**Type:** string

**Value:** none | warning | error

**Default:** warning

## Overspecified targets in kinematic loops

Select the diagnostic action to take if the application detects overspecified targets contained in kinematic loops in the model.

### Settings

**Default:** warning

none

The application does not check for this situation.

warning

When the application detects this situation, it displays a warning.

error

When the application detects this situation, it terminates the simulation and displays an error message.

### Command-Line Information

**Parameter:** SimMechanicsJointTargetOverSpecification

**Type:** string

**Value:** none | warning | error

**Default:** error

# SimMechanics Pane: Explorer

Open Mechanics Explorer on model update or simulation

## Open Mechanics Explorer on model update or simulation

Start Mechanics Explorer when model is updated or simulated.

### Settings

**Default:** on

On

Model Explorer starts when model is updated or simulated.

Off

Model Explorer does not start when model is updated or simulated.

### Tip

If you clear this check box, you can start Model Explorer by selecting **Desktop > Mechanics Explorers** from the MATLAB® Command Window.

### Command-Line Information

**Parameter:** SimMechanicsOpenEditorOnUpdate

**Type:** string

**Value:** 'on' | 'off'

**Default:** 'on'



# SimMechanics Visualization

---

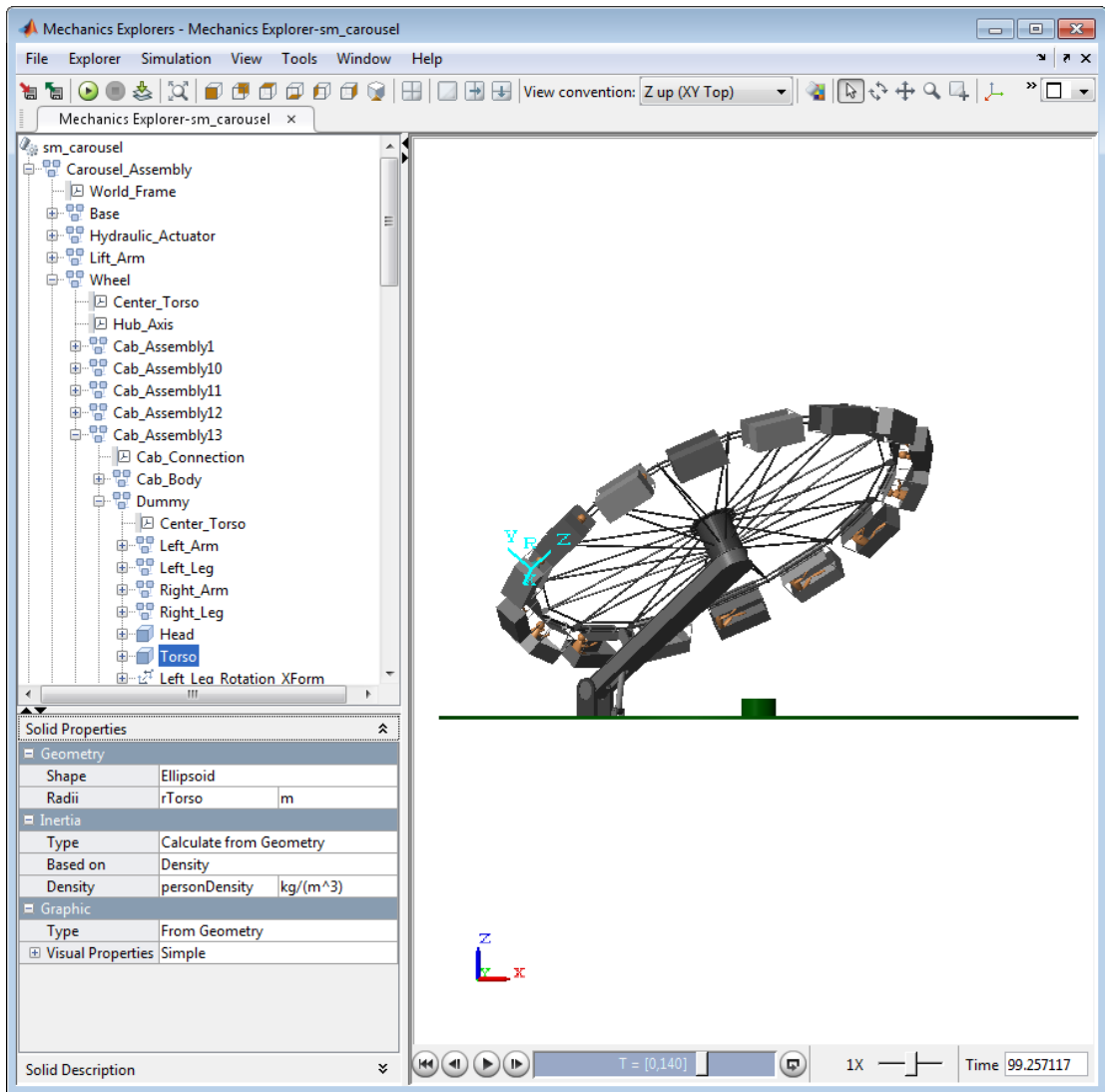
- “Visualization with Mechanics Explorer” on page 3-2
- “Menu and Tool Bars” on page 3-8
- “Tree View and Property Panes” on page 3-11
- “Context Menu Display Controls” on page 3-13
- “Animation Toolstrip” on page 3-14

## Visualization with Mechanics Explorer

In this section...
“Introduction to Mechanics Explorer” on page 3-2
“Visualization Requirements” on page 3-4
“Configure Mechanics Explorer for Automatic Start-Up” on page 3-4
“Open Mechanics Explorer” on page 3-4

### Introduction to Mechanics Explorer

Mechanics Explorer is a tool used to visualize and animate mechanical models in SimMechanics. This tool provides you the entire visualization functionality available through SimMechanics. With it, you can visualize your model in its initial state and simulate it as a function of time.



Mechanics Explorer is the primary visualization tool of SimMechanics. This tool contains the following panes:

Pane	Purpose
Visualization pane	Displays a 3-D representation of a SimMechanics model
Tree view pane	Displays the model component hierarchy
Property pane	Displays selected component properties

## Visualization Requirements

To visualize a model in SimMechanics, you must have Silicon Graphics OpenGL<sup>®</sup> installed in your system.

## Configure Mechanics Explorer for Automatic Start-Up

By default, when you update or simulate a model, SimMechanics opens a new Mechanics Explorer window. It is in this window that SimMechanics displays your model. If the configuration settings have changed, you may need to reconfigure Mechanics Explorer for automatic start-up. To do this:

- 1 In the Simulink Editor window containing your model, select **Simulation > Model Configuration Parameters**.  
  
Parameters
- 2 In the Configuration dialog box, select **SimMechanics 2G > Explorer**.
- 3 Check the **Open Mechanics Explorer on model update or simulation** box.

Mechanics Explorer is now configured for automatic start-up. Updating or simulating a mechanical model from the Simulink Editor window causes a Mechanics Explorer window to open.

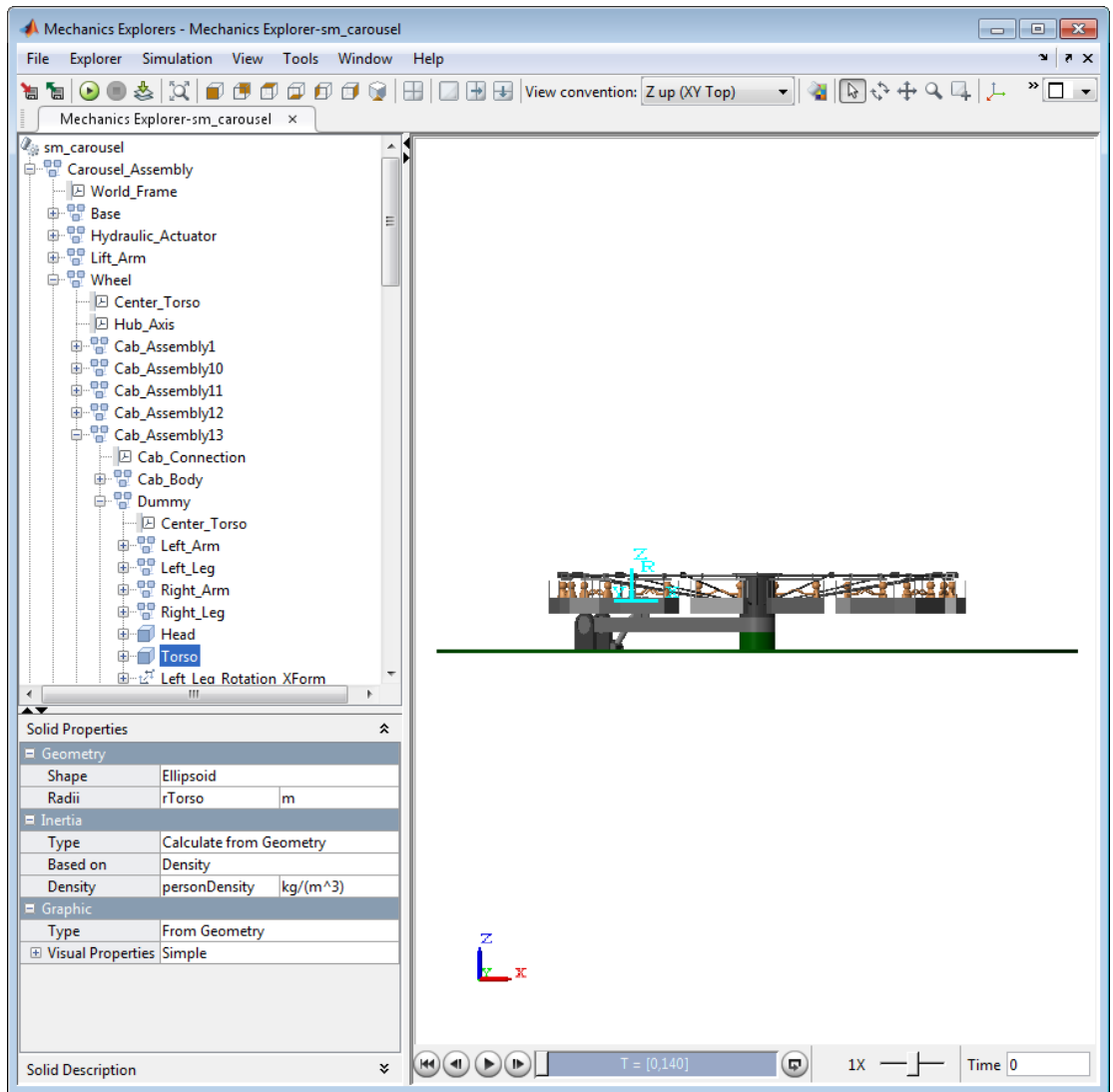
## Open Mechanics Explorer

With Mechanics Explorer configured to start automatically, you do not have to manually open a Mechanics Explorer window. Simply follow the procedure for updating or simulating a model and, if a Mechanics Explorer window is not yet open, a new window opens up.


**Open Mechanics Explorer for Model Update**

- 1 In the Simulink Editor window containing your model, select **Simulation > Update Diagram**. Alternatively, on your keyboard, press **Ctrl+D**.
- 2 A Mechanics Explorer window opens, displaying your model in its initial state.

### 3 SimMechanics Visualization



### Open Mechanics Explorer for Model Simulation

- 1 In the Simulink Editor window containing your model, select **Simulation > Run**. Alternatively, on your keyboard press Ctrl+T or, in the Simulink Editor toolbar, press the **Run** button .
- 2 A Mechanics Explorer window opens. The window displays a 3-D animation of the model.

---

**Note:** Each Mechanics Explorer window is named after the model it displays. E.g. Mechanics Explorer-*sm\_four\_bar*.

---

## Menu and Tool Bars

### In this section...

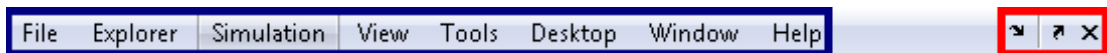
“Menu Bar” on page 3-8

“Toolbar” on page 3-9

Mechanics Explorer provides you menu and tool bars for easy access to tools, including simulation and visualization tools. These two bars contain the majority of the tools which you can use with SimMechanics. Some of the tools provided by the Menu and Tool bars are also accessible in the context-sensitive menu, which you can open by right-clicking the visualization window in Mechanics Explorer.

### Menu Bar

The Menu bar contains eight menu items, on the left end of the bar, and three desktop control icons, on the right end of the bar.






### Menu Items

Menu Item	Function
File	Open and close a file, import and save data
Explorer	Save and close a Mechanics Explorer window
Simulation	Start and stop a simulation
View	Select and modify a model viewpoint
Tools	Troubleshoot a model and record an animation
Desktop	Dock and undock a Mechanics Explorer window, show and hide the Mechanics Explorer toolbar
Window	Configure Mechanics Explorer window






Menu Item	Function
Help	Search MathWorks documentation for Mechanics Explorer or other product help


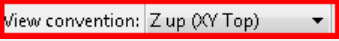




### Desktop Control Icons

Desktop Control Icon	Function
Dock Mechanics Explorers 	Open Mechanics Explorer window inside the MATLAB window
Undock Mechanics Explorers 	Open Mechanics Explorer outside the MATLAB window
Close Mechanics Explorers 	Close active Mechanics Explorer window

### Toolbar

The Tool bar contains nine sections.

Toolbar Icons	Function
	<ul style="list-style-type: none"> <li>• Save Mechanics Explorer configuration to model</li> <li>• Restore Mechanics Explorer configuration from model</li> </ul>
	<ul style="list-style-type: none"> <li>• Start simulation</li> <li>• Stop simulation</li> <li>• Update model block diagram</li> </ul>
	<ul style="list-style-type: none"> <li>• Fit to view</li> <li>• Front view</li> <li>• Back view</li> <li>• Top view</li> <li>• Bottom view</li> <li>• Left view</li> <li>• Right view</li> <li>• Isometric view</li> </ul>

Toolbar Icons	Function
	<ul style="list-style-type: none"> <li>• Split the visualization pane into four screens</li> <li>• Use a single full-sized visualization window</li> <li>• Split the visualization pane vertically into two screens</li> <li>• Split the visualization pane horizontally into two screens</li> </ul>
	<ul style="list-style-type: none"> <li>• Change default frame orientation. Choices include Z Up, Z Down, and Y up.</li> </ul>
	<ul style="list-style-type: none"> <li>• Change background color of visualization window</li> <li>• Select model component</li> <li>• Rotate model</li> <li>• Pan model</li> <li>• Zoom model</li> <li>• Zoom model region</li> </ul>
	<ul style="list-style-type: none"> <li>• Show model frames</li> <li>• Show model Center-of-Masses</li> </ul>
	<ul style="list-style-type: none"> <li>• Record movie of animation</li> </ul>
	<ul style="list-style-type: none"> <li>• Tile multiple Mechanics Explorer windows into a four-matrix</li> <li>• Tile multiple Mechanics Explorer windows into two vertical windows</li> <li>• Tile multiple Mechanics Explorer windows into two horizontal windows</li> <li>• Maximize active Mechanics Explorer window to occupy entire region of visualization window</li> </ul>

# Tree View and Property Panes

**In this section...**

“Introduction to Tree View and Property Panes” on page 3-11

“Browse Model” on page 3-11

Mechanics Explorer contains **Tree View** and **Property** panes so that you can navigate a model and explore model parameters.

## Introduction to Tree View and Property Panes

By default, Mechanics Explorer displays both a **Tree View** pane and a **Property** pane. With these panes, you can navigate the subsystem and block hierarchy of a model. You can also view the properties of any block in the model.

The **Tree View** browser helps you navigate a mechanical model by organizing blocks into a hierarchical structure. You can select a subsystem in your model, and examine any of its constituent blocks. If a subsystem is in turn built from other subsystems, you can select the new subsystem of interest, and navigate to any of its constituent blocks, all from the model browser. You can also expand any single block to examine any of its frames.

The **Property** pane displays the properties of the component that you selected in the **Tree View** browser. If the component is a block, the pane displays the dialog box parameters for that block. If the component is a frame port, the pane displays other frame ports that directly connect to the selected frame port.

## Browse Model

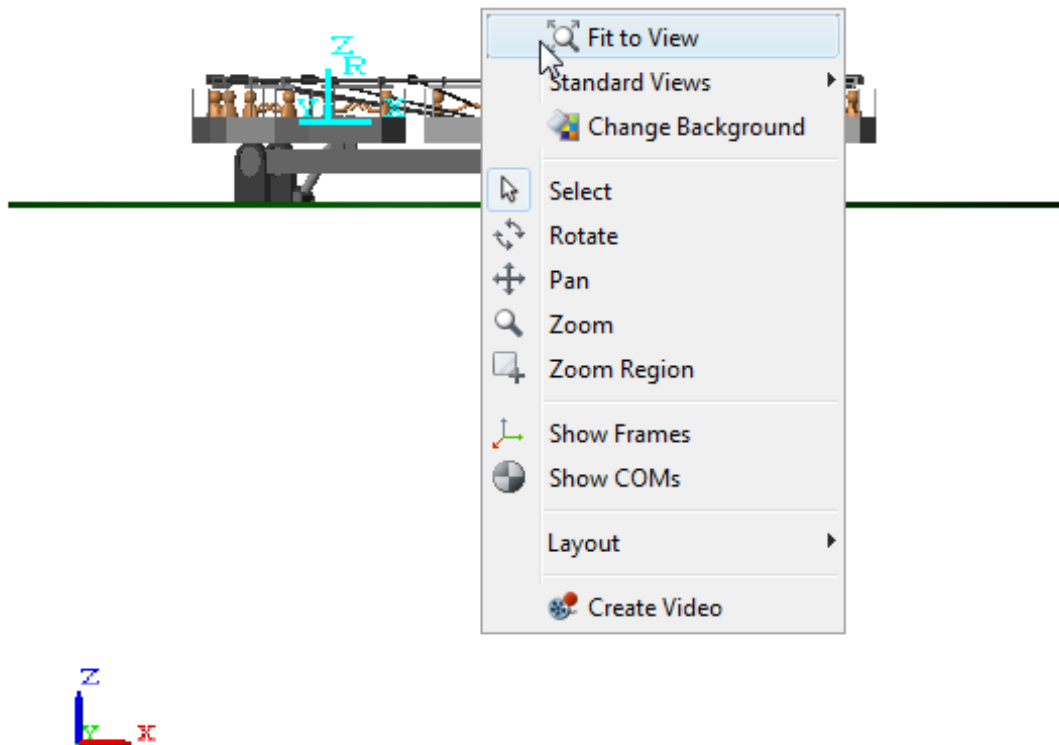
Use the **Tree View** pane to highlight a component and its associated frames in the visualization window, or to view its properties:

- 1 In the **Tree View** pane, select the component you wish to analyze:
  - The selected component and all associated frames are highlighted in the visualization window
  - The properties of the selected component are displayed in the property viewer
- 2 Click the expandable menu for the selected component, displaying its frames.

- 3 Select the frame you wish to examine:
  - Only the selected frame is highlighted in the visualization window
  - All frames which connect to the selected frame are highlighted in the property viewer

## Context Menu Display Controls

You can right-click the visualization pane to open a context menu, giving you quick access to commonly used visualization and animation tools. The following figure shows an open context menu in a Mechanics Explorer window. You can also access each of these tools from the Mechanics Explorer toolbar. For more information about context menu options, see “Menu and Tool Bars” in this reference documentation.



## Animation Toolstrip

### In this section...

“Requirements for Animation Recording and Playback” on page 3-14

“How Animation and Playback Work” on page 3-14

“Play Control Buttons” on page 3-15

“Slider and Loop Controls” on page 3-15

“Playback Speed Control” on page 3-16

“Playback Time Counter” on page 3-16

### Requirements for Animation Recording and Playback

- With SimMechanics visualization, you record simulation animations in Microsoft Audio Video Interleave® (AVI) format.
  - Animations are recorded as compressed AVI files by the MATLAB `VideoWriter` class using the default Motion JPEG codec, which is available on most operating systems.
  - SimMechanics visualization does not support uncompressed AVI recording.
- To play back AVI files, you need an AVI-compatible media application.
  - You can use an external AVI-compatible player.
  - Alternatively, use the MATLAB `VideoReader` class and its `read` method instead.

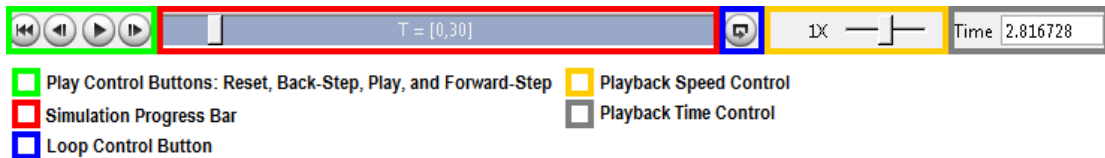
### How Animation and Playback Work

Animation is cached during model simulation. What you see during simulation is the animation playback, unless the simulation is slower than the animation caching. In that case, the animation goes no faster than the simulation can produce the cache.

Once a partial or complete animation is cached from simulation, starting the animation again plays back the cache, without running the simulation a second time.

- While simulating for the first time, you can go back to previous animation time steps (already cached), but not forward in time.
- When playing back second, third, etc., time, you can go backward and forward to any time in the cached animation.

- Animation is cached on a per-session basis. This cache is lost when you close your visualization window session.
- Creating a video as a permanent record on your file system outside of MATLAB saves the cached animation permanently.



## Play Control Buttons

Use the play control buttons to play and stop an animation, skip a step, or reset it to the beginning:

Play Control Button	Description
Start/Stop	Start/stop the cached animation
Reset	Move the animation playback time back to zero
Forward Step	Skip forward in time
Back Step	Skip backward in time

## Slider and Loop Controls

### Slider Control

The Slider control moves the animation playback time to an arbitrary point in the animation. The Playback time counter displays the resulting playback time.

- If the animation is not running, moving the Slider control changes the animation time from one fixed time to another.
- If the animation is running, moving the Slider control changes the animation time to whatever time you stop the slider. But the animation then continues running from that time.

Without manual control, the Slider control indicates how far in relative time the cached animation has played.

### Loop Control

You use the loop control button to automatically replay an animation from the start once it reaches the end. When you click this button, the cached animation replays indefinitely until you click the **Stop** button.

The Loop control is disabled by default. Enable looping by clicking this button. Disable looping by clicking it again.

### Simulation Progress Bar

The Simulation progress bar indicates how far in time the model simulation has proceeded, as opposed to how far in time the cached animation has played.

Fast models complete simulation before the cached animation is finished playing. Slower models cache and play the animation at a speed limited by how fast the simulation proceeds.

### Playback Speed Control

The Playback speed control adjusts the relative speed at which the animation plays back, as compared to the default speed of 1. The relative playback speed is incremented by multiples of 2 and 1/2.

Moving the Playback speed control to the left slows the playback speed, by a factor of 1/2, 1/4, etc. Moving the Playback speed control to the right speeds up the playback speed, by a factor of 2, 4, etc.

### Playback Time Counter

The Playback time counter is both an indicator and a controller of animation playback time. A valid playback time is one that lies between 0 and the full simulation time, inclusive.

- During animation, or stopped at an intermediate time between start and end, the Playback time counter displays the current or instantaneous animation time.
- Entering a valid animation time in the Playback time counter field and pressing Enter at the keyboard moves the animation playback to the animation frame closest to whatever time you entered.



# Functions – Alphabetical List

---

### **smnew**

Open new model with common SimMechanics blocks

### **Syntax**

```
smnew
smnew(modelname)
smnew(modelname,solver)
```

### **Description**

`smnew` creates a new SimMechanics model, with required and commonly used blocks already on the model canvas, and opens the `sim_lib` library. The default model name is `untitled` and the default solver is `ode45` (the recommended solver).

The new model contains the following blocks:

- World Frame
- Solid
- Rigid Transform
  
- Solver Configuration
- Mechanism Configuration
- PS-Simulink Converter
- Simulink-PS Converter
- Scope

`smnew(modelname)` creates a new SimMechanics model with the specified name. If you specify an invalid model name, the model name is `untitled`. The default solver is `ode45`, which is the recommended solver.

`smnew(modelname,solver)` creates a new SimMechanics model with the specified name and Simulink solver type. If you do not specify a model name, then the model name is `untitled`.

## Input Arguments

### **modelName**

String specifying the name for the new model. Do not include a file extension.

**Default:** untitled

### **solver**

String specifying the name of a Simulink solver. If you choose not to use the default `ode45` solver, other recommended solver types for SimMechanics models are `ode15s`, `ode23t`, and `ode14x`. You can use other Simulink solvers, but, depending on the particular model, they may be less suitable. For more information, see “Improving Performance” and “Setting Up Solvers for Physical Models”.

**Default:** `ode45`.

## Examples

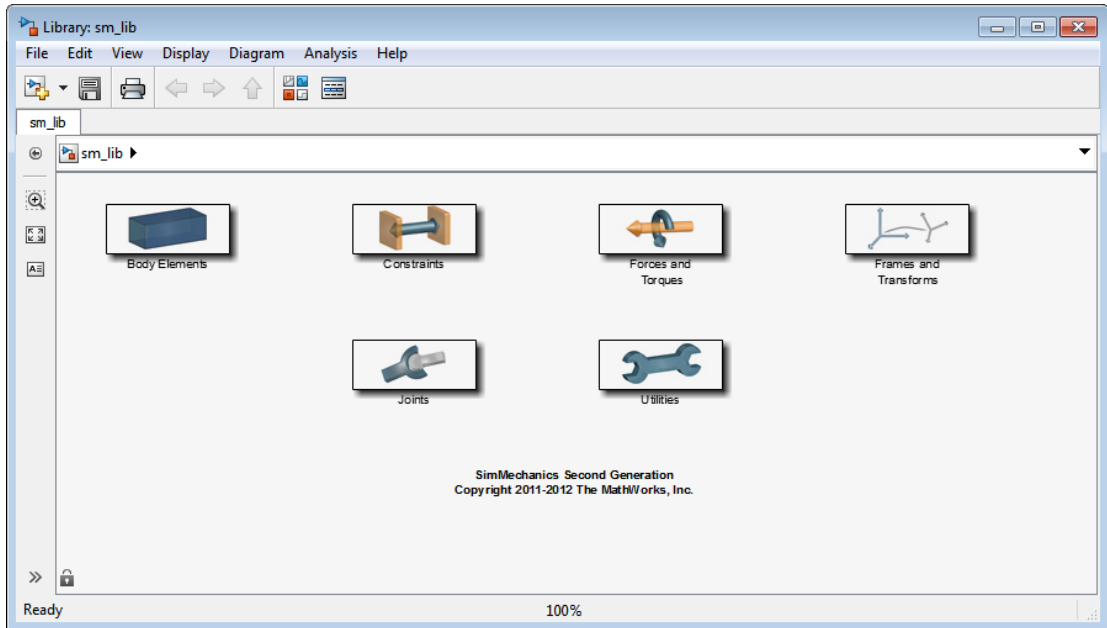
### **New SimMechanics Model with Defaults**

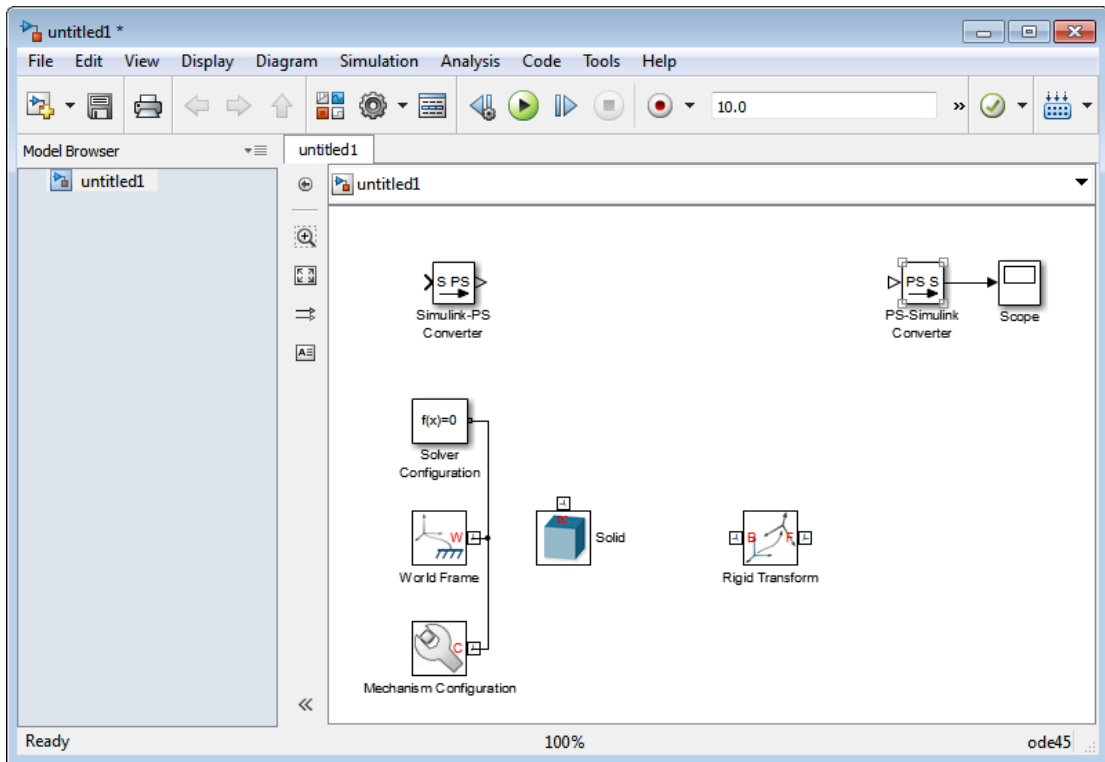
Create a new SimMechanics model, using the default model name (`untitled`) and the default solver (`ode45`).

```
smnew
```

The command opens the `sm_lib` library, as well as a SimMechanics model called `untitled`, with several commonly used blocks.

## 4 Functions—Alphabetical List



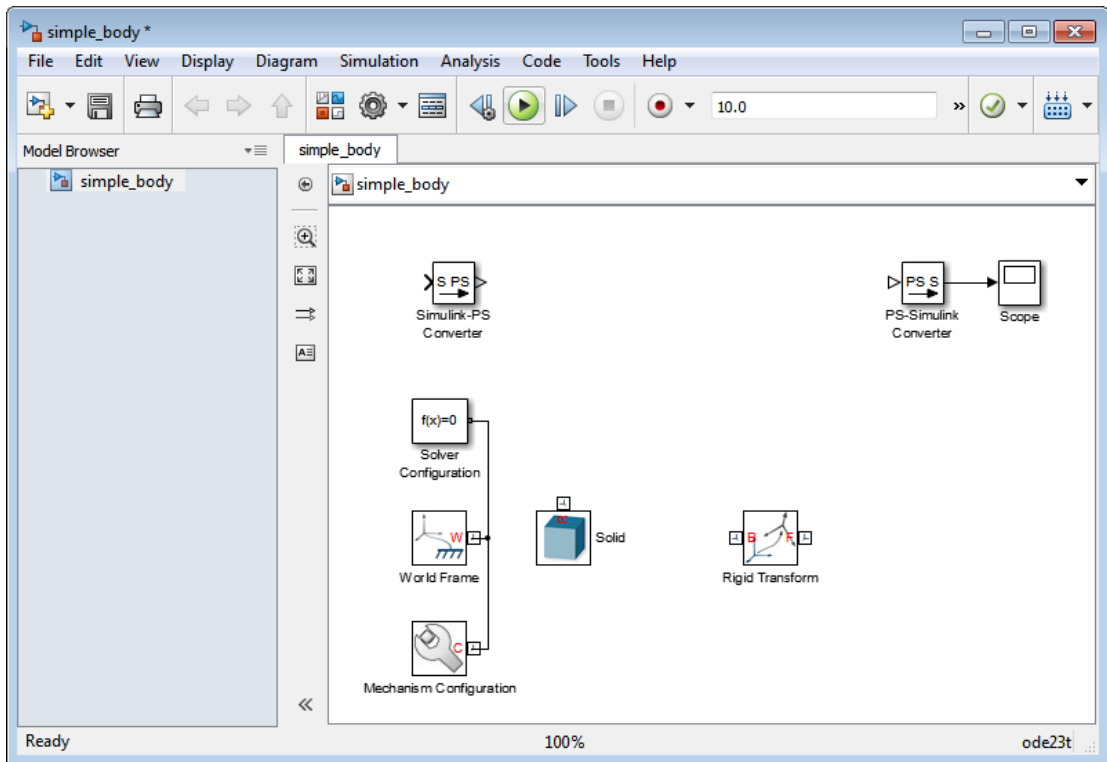


### Named New SimMechanics Model with custom Solver

Create a new SimMechanics model, named `simple_body`. Use the `ode23t` solver.

```
sm_new('simple_body', 'ode23t')
```

The command opens the `sm_lib` library, as well as a SimMechanics model called `simple_body`, with several commonly used blocks. The model uses the `ode23t` solver.



### See Also

sm\_lib | smimport

# smimport

Generate SimMechanics Second Generation model from SimMechanics Import XML file

## Syntax

```
H = smimport('filename')  
H = smimport('filename', 'modelName', 'modelname')
```

## Description

`H = smimport('filename')` imports SimMechanics XML file `filename.xml` into a new Simulink model with the same name. SimMechanics generates the new model according to XML file specification, adding and connecting any required blocks. For CAD-generated XML files, the new model represents a CAD assembly.

`filename` is a string specifying the XML file name. This string must be a valid model name, using any combination of letters, digits, and underscores such that the first character is a letter and the length of the string is smaller than `namelengthmax`. If `filename` is invalid, SimMechanics generates a valid name based on `filename`.

If a model with the same name is open, `smimport` returns a warning and appends an integer to `filename` in order to create a unique model name. If a model with the same name exists in the MATLAB workspace or path, `smimport` returns a warning asking you to change the model name.

`filename` supports path specification relative to the MATLAB working directory. Alternatively, `filename` can contain the full path to the corresponding XML file. In the absence of a path, `smimport` searches for `filename` inside the MATLAB path.

By default, `smimport` assumes `filename` refers to an XML file in the MATLAB workspace or path. `filename` does not require an explicit XML extension. If `filename` is not an XML file, or if it is not in the MATLAB workspace or path, `smimport` returns an error.

`H = smimport('filename', 'modelName', 'modelname')` imports XML file `filename` into a new SimMechanics model with name `modelname`.

## Input Arguments

### **filename**

String specifying input XML file name. Replace *filename* with the XML file name you want to import. The `.xml` extension is optional.

This input is required.

**Default:** ''

## Name-Value Pair Arguments

Specify optional comma-separated pairs of Name, Value arguments. **Name** is the argument name and **Value** is the corresponding value. **Name** must appear inside single quotes (' '). You can specify several name and value pair arguments in any order as `Name1, Value1, . . . , NameN, ValueN`.

### **'ModelName'**

String specifying generated SimMechanics model name.

This input is optional.

**Default:** ''

## Examples

Use the `smimport` command to automatically generate a SimMechanics model with default or custom names.

### Import Robot Assembly with Default Name

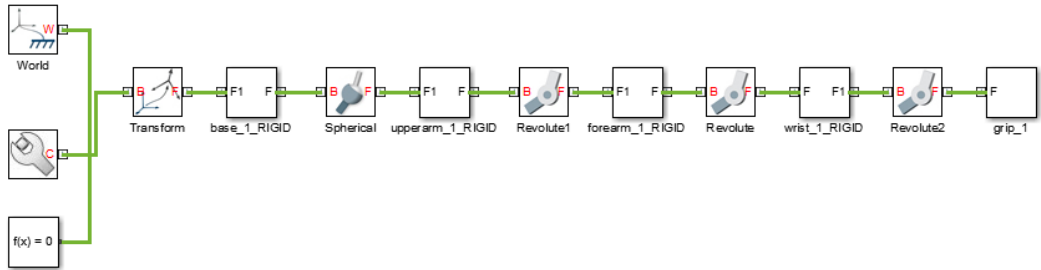
SimMechanics provides a SimMechanics Import XML file that you can use to import a robot arm model. The file, named `sm_robot.xml`, provides SimMechanics the data it needs to automatically generate the model. You can generate a SimMechanics Import XML file from a supported CAD platform using the SimMechanics Link utility.

To import the `sm_robot` assembly into SimMechanics:



- 1 At the command line, enter

```
smimport('sm_robot.xml');
```



- 2 Save the file in a convenient folder.

## Import Robot Assembly with Custom Name

You can specify a custom model name directly at the MATLAB command line. You accomplish this task by including a `ModelName` argument in the `smimport` command. For example, to assign the `robot_arm` name to the `sm_robot` assembly, at the MATLAB command line, enter:

```
smimport('sm_robot', 'ModelName', 'robot_arm');
```

SimMechanics automatically generates a robot arm model with the `robot_arm` name.

## More About

### Tips

- `smimport` provides CAD integration with SimMechanics. Use SimMechanics Link™ to generate a valid SimMechanics XML file from SolidWorks™, Creo™ (Pro/Engineer™), or AutoCAD Inventor™. Then, use `smimport` to generate a corresponding SimMechanics model.

### See Also

`sm_lib` | `smnew`

## sm\_lib

Open the SimMechanics block library

### Syntax

sm\_lib

### Description

sm\_lib opens the SimMechanics block library in a separate window. Double-click a sublibrary to view the blocks that it provides. The table summarizes the purpose of each sublibrary.

Sublibrary	Use to Model
Body Elements	Rigid body geometry, inertia, and graphical appearance
Constraints	Kinematic constraints on the relative motion of two rigid body frames
Forces and Torques	Generic and specialized forces and torques acting on or between rigid body frames
Frames and Transforms	Frames and their spatial relationships
Gears and Couplings	Specialized gear constraints between rigid bodies
Joints	Connections between rigid bodies with desired translational and rotational degrees of freedom
Utilities	Mechanical and simulation-specific parameters for an entire mechanism

## More About

### Tips

- Use `sm_lib` to open the SimMechanics block library from the SimMechanics command line.
- `sm_lib` loads only the SimMechanics block library, so you do not have to wait for other Simulink libraries to load.

### See Also

`smimport` | `smnew`

