Simscape™ Fluids™

User's Guide
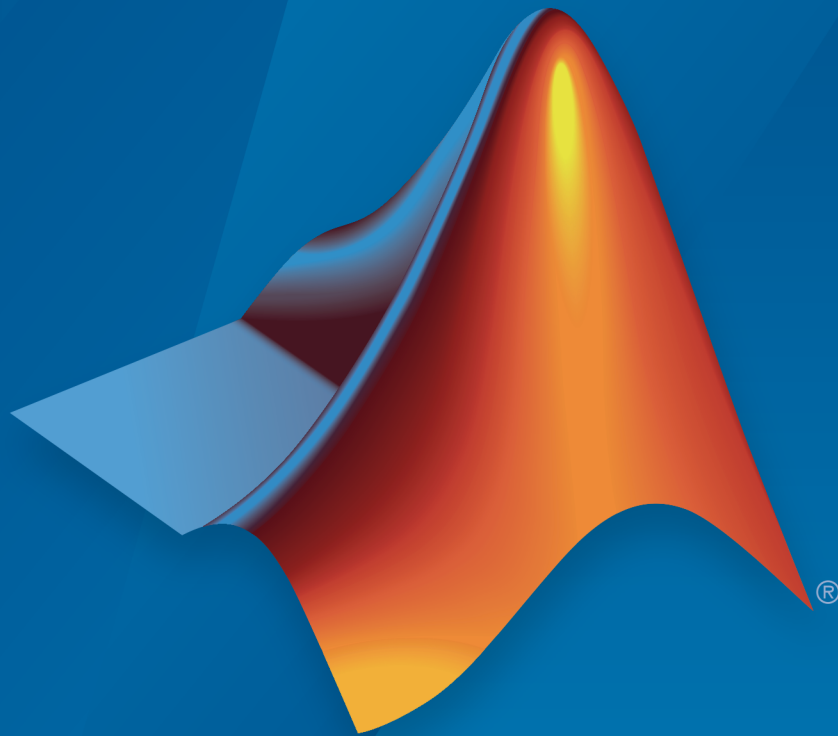
MATLAB&SIMULINK®

MathWorks®

How to Contact MathWorks

Latest news:	www.mathworks.com
Sales and services:	www.mathworks.com/sales_and_services
User community:	www.mathworks.com/matlabcentral
Technical support:	www.mathworks.com/support/contact_us
Phone:	508-647-7000

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

Contents

Getting Started

Simscape Fluids Product Description

Model and simulate fluid systems

Simscape Fluids provides component libraries for modeling and simulating fluid systems. It includes models of hydraulic pumps, valves, actuators, pipelines, and heat exchangers. You can use these components to develop fluid power systems such as front-loader, power steering, and landing gear actuation systems. Simscape Fluids also enables you to develop engine cooling, gearbox lubrication, and fuel supply systems. You can integrate mechanical, electrical, thermal, and other physical systems into your model using components from the Simscape family of products.

Simscape Fluids helps you develop control systems and test system-level performance. You can create custom component models with the MATLAB® based Simscape language, which enables text-based authoring of physical modeling components, domains, and libraries. You can parameterize your models using MATLAB variables and expressions, and design control systems for your hydraulic system in Simulink®. To deploy models to other simulation environments, including hardware-in-the-loop (HIL) systems, Simscape Fluids supports C-code generation.

Key Features

- Pump models, including centrifugal, jet, and axial-piston pumps
- Valve models, including check valves, directional valves, and pressure relief valves
- Rotational and translational actuator models, including optional friction and centrifugal forces
- Heat exchanger models for modeling heating and cooling systems
- Tank and pipe models with elevation effects for modeling fluid transportation systems
- Customizable library of common fluids, with ability to import properties from common databases
- MATLAB based Simscape language for creating custom component models
- Support for C-code generation (with Simulink Coder™)

Simscape Fluids Applications

Simscape Fluids software enables you to model a variety of thermal and isothermal liquid systems. Key application areas include fluid transport, power generation, as well as heating and cooling. Simscape Fluids blocks are organized into three libraries spanning two Simscape liquid domains, Hydraulics and Thermal Liquid. The libraries include:

- Hydraulics — Model liquid systems at constant temperature, without regard for heat transfer. The fluid properties are constant functions of temperature. This library is suitable for systems where temperature fluctuations are small, as in hydraulic actuators operating at nearly constant temperature.

- Thermal Liquid — Model liquid systems at variable temperature, with thermal effects due to convective and conductive heat transfer. The fluid properties are functions of pressure and temperature. This library is suitable for systems where temperature fluctuations can be large, as in engine oil coolers.

- Fluid Network Interfaces — Model heat exchange and other physical interactions between otherwise distinct liquid networks. The library includes convenience blocks such as Interface (TL-IL) that you can use to expand any existing Hydraulics models with Thermal Liquid components.

Start a New Simscape Fluids Model

You can use the Simscape model template as a starting point for your Simscape Fluids models. The template includes the required `Solver Configuration` block and the commonly used `Simulink-PS Converter` and `PS-Simulink Converter` blocks.

To open the Simscape model template, at the MATLAB command prompt, enter `ssc_new`. Simscape software opens the model template along with the Foundation library. Enter `SimscapeFluids_lib` to open the Simscape Fluids block library.

The figure shows the Simscape model template. Drag blocks from the Simscape and Simscape Fluids libraries to represent the various system components. Use specialized blocks from physical modeling products such as Simscape Multibody™ and Simscape Driveline™ to model multiphysics systems.
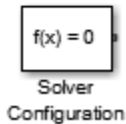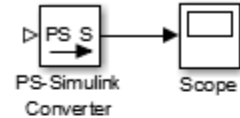
Simulating Hydraulic Models

Simulation Basics

Simscape Fluids software gives you multiple ways to simulate and analyze hydraulic power and control systems in the Simulink environment. Running a hydraulic simulation is similar to running a simulation of any other Simscape model. See "Desktop Simulation" in the Simscape documentation for a discussion of the following topics:

- Explanation of how Simscape Fluids software validates and simulates a model
- Specifics of using Simulink linearization commands in Simscape Fluids models
- Generating code for Simscape Fluids models
- Restrictions and limitations on using Simulink tools in Simscape Fluids models

All these aspects of simulating Simscape Fluids models are exactly the same as for Simscape models.

Selecting a Solver

Simscape Fluids software supports all of the continuous-time solvers that Simscape supports. For more information, see "Setting Up Solvers for Physical Models" in the Simscape documentation.

You can select any of the supported solvers for running a Simscape Fluids simulation. The variable-step solvers, `ode23t` and `ode15s`, are recommended for most applications because they run faster and work better for systems with a range of both fast and slow dynamics.

To use Simulink Coder software to generate standalone C or C++ code from your model, you must use the `ode14x` solver. For more information about code generation, see "Code Generation" in the Simscape documentation.

Specifying Simulation Accuracy/Speed Tradeoff

To trade off accuracy and simulation time, adjust one or more of the following parameters:

- **Relative tolerance** (in the Configuration Parameters dialog box)
- **Absolute tolerance** (in the Configuration Parameters dialog box)
- **Max step size** (in the Configuration Parameters dialog box)
- **Constraint Residual Tolerance** (in the Solver Configuration block dialog box)

In most cases, the default tolerance values produce accurate results without sacrificing unnecessary simulation time. The parameter value that is most likely to be inappropriate for your simulation is **Max step size**, because the default value, `auto`, depends on the simulation start and stop times rather than on the amount by which the signals are changing during the simulation. If you are concerned about the solver missing significant behavior, change the parameter to prevent the solver from taking too large a step.

The Simulink documentation describes the following parameters in more detail and provides tips on how to adjust them:

- "Relative tolerance"
- "Absolute tolerance"
- "Max step size"

The `Solver Configuration` block reference page in the Simscape documentation explains when to adjust the **Constraint Residual Tolerance** parameter value.

Nonphysical Values During Simulation

Sometimes Simscape Fluids models may display physically unattainable values during simulation because of an irregularity within the model. In general, simulation does not stop if one or more variables assume nonphysical values, such as gauge pressures below –1 bar, negative value of fluid volume in the reservoir, and so on. The rationale is that, at the end of simulation, the user knows the extent of the irregularity rather than just the fact that the reservoir is short of fluid or there is a pressure drop below vaporization level.

If you see that your model displays physically unattainable values during simulation, you must analyze your model and iteratively modify your design to smooth out the irregularities.

You can set your models to either warn or stop simulating with an error when absolute pressure in a hydraulic chamber falls below absolute zero. The default behavior is to stop simulation with an error. You can change this by using the `Hydraulic Fluid` block or the `Custom Hydraulic Fluid` block connected to the loop, and have the simulation continue with a warning. See the block reference pages for details.

Troubleshooting Hydraulic Models

Simscape Fluids simulations can stop before completion with one or more error messages. For a discussion of generic error types and error-fixing strategies, see "Troubleshooting Simulation Errors" in the Simscape documentation. The following troubleshooting techniques are specific to Hydraulic models:

- Review the model configuration. If your error message contains a list of blocks, look at these blocks first. Also look for:

    - Wrong connections — Verify that the model makes sense as a hydraulic system. For example, look for accumulators connected to the pump outlet without check valves; cylinders connected against each other, so that they try to move in opposite directions; directional valves bypassed by a huge orifice, and so on.

    - Wrong use of hydraulic elements — Simscape Fluids blocks model their respective hydraulic units within certain limits. For example, an Ideal Pressure Source block can simulate a pump only when the pressure remains constant (see "Modeling Power Units" on page 2-44). Similarly, the Pressure Relief Valve block is a steady-state representation of a real valve. A block may exhibit wrong behavior if it is placed in the wrong environment. Always check the validity of the model for a particular environment and the simulation objectives.

- Avoid portions of the system getting isolated from the main system. An isolated or "hanging" part of the system could affect computational efficiency and even cause failure of computation. Use the **Leakage Area** parameter, introduced specifically for this purpose, to maintain numerical integrity of the circuit. This parameter is present in all the directional valve blocks, pressure control and flow control valve blocks, and most of the variable orifices.

- Avoid "dry" nodes in a hydraulic system. By adding a hydraulic chamber to a node, you can considerably improve the convergence and computational efficiency of a model. Adding a chamber adds a degree of freedom. By adding a chamber, you replace a complex algebraic constraint (the dry node) by a dynamic constraint. In the Mechanical domain, this is like replacing an ideal gearbox with a gearbox that includes compliance (spring and damper). The hydraulic chamber is represented

by the `Constant Volume Hydraulic Chamber` block in the Simscape Hydraulic Elements library.

MathWorks recommends that you build, simulate, and test your model incrementally. Start with an idealized, simplified model of your system, simulate it, verify that it works the way you expected. Then incrementally make your model more realistic, factoring in effects such as fluid compressibility, fluid inertia, and the other things that describe real-world phenomena. Simulate and test your model at every incremental step. Use subsystems to capture the model hierarchy, and simulate and test your subsystems separately before testing the whole model configuration. This approach helps you keep your models well organized and makes it easier to troubleshoot them.

Sharing Models with Simscape Users

The Simscape Editing Mode functionality lets you open, simulate, and save models that contain blocks from add-on products, including Simscape Fluids blocks, in Restricted mode, without checking out add-on product licenses, as long as the products are installed on your machine.

This functionality allows a user, *model developer*, to build a model that uses Simscape and Simscape Fluids blocks and share that model with other users, *model users*. When building the model in Full mode, the model developer must have both a Simscape license and a Simscape Fluids license. Once the model is built, model users need only to check out a Simscape license to simulate the model and fine-tune its parameters in Restricted mode. As long as no structural changes are made to the model, model users can work in Restricted mode and do not need to check out Simscape Fluids licenses.

Another workflow lets multiple users, who all have Simscape licenses, share a small number of Simscape Fluids licenses by working mostly in Restricted mode, and temporarily switching models to Full mode only when they need to perform a specific design task that requires being in Full mode.

For a complete description of this functionality, see "Add-On Product License Management" in the Simscape documentation.

2

Modeling Hydraulic Systems

Simscape Fluids Modeling Principles

Block Diagrams

Simscape Fluids block diagrams are like other Simscape models. They represent fluid systems by means of blocks and connection lines. Each block represents a physical component and its salient attributes. Each line represents a physical connection or interaction between components—including any exchanges in mass, energy, and momentum. Lines connect through ports representing inlets, outlets, and other component features.
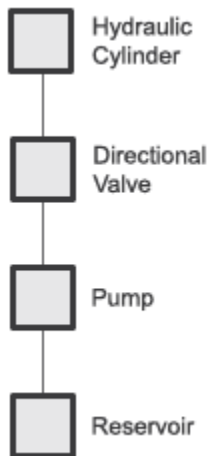
Blocks as Components and Lines as Connections

You can model fluid components using blocks from the Simscape Foundation or Simscape Fluids libraries. Simscape Foundation blocks represent idealized components such as local restrictions, mechanical converters, and constant-volume chambers. Simscape Fluids blocks represent specialized components with detailed component physics, such as directional valves, single-acting actuators, and gas-charged accumulators.

Like the systems they represent, Simscape Fluids models are often multi-domain in nature. The hydraulic actuator model on page 2-2 is one example. The model contains a hydraulic network representing the hydraulic actuation system and a Mechanical Translational network representing the actuator load. To model components from different domains, you can use blocks from the various Simscape Foundation libraries or from the several specialized physical modeling products.

Connection and Signal Lines

Lines can join blocks through physical conserving ports, ▫, or physical signal ports, △. A line between physical conserving ports represents a physical connection bound by physical conservation principles. A line between physical signal ports represents a component input, such as a change in valve position, or sensing output, such as a fluid pressure measurement.
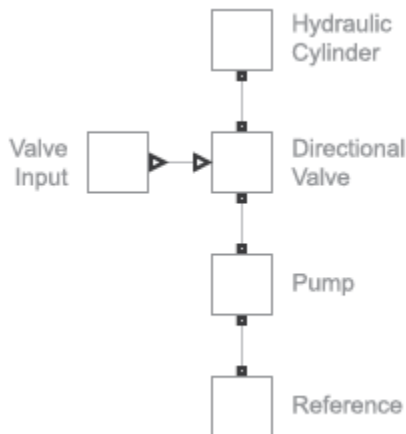
Conserving and Physical Signal Ports

Physical connection lines are nondirectional. They apply constraints between the Across and Through variables of the connected conserving ports, but do not transmit numerical

values in any particular direction. Physical signal lines are directional. They pass numerical signals and associated physical units in a specific direction from one block to another, often as block inputs or outputs.

Blocks and physical connection lines are color-coded by physical domain. The various colors help to differentiate the physical domains comprising a Simscape model. Use them as visual guides when deciding if a physical connection between components is valid. All physical connection lines must join conserving ports of the same domain. The figure shows the color legend for the physical domains associated with Simscape Fluids blocks.

Color	Domain
———	Hydraulic
———	Thermal Liquid
———	Thermal
———	Mechanical Rotational
———	Mechanical Translational

Simscape Fluids Color Coding

For more information, see "Domain-Specific Line Styles".

Conserving and Signal Ports

Ports always connect to like ports. A physical conserving port connects only to other physical conserving port—and only if they belong to the same physical domain. Conserving ports are associated with specific Across and Through variables and cannot connect to conserving ports associated with different variables. You can directly connect a Hydraulic conserving port to another Hydraulic conserving port but not to a Thermal Liquid conserving port.

Hydraulic-Thermal Liquid Invalid Connection

Single-Acting Actuator (TL)

3-Way Directional Valve
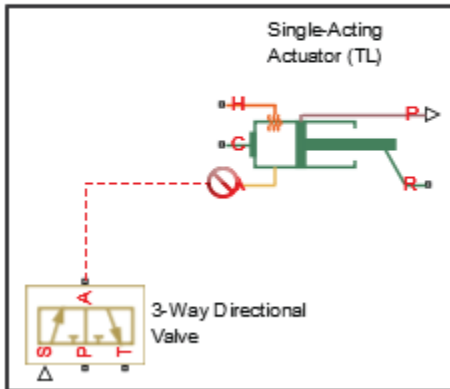
Thermal Liquid-Thermal Liquid Valid Connection

Single-Acting Actuator (TL)

3-Way Directional Valve (TL)
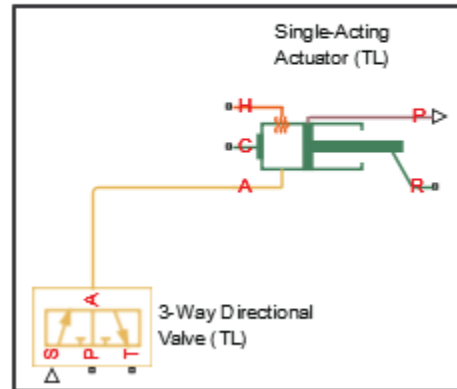
To interconnect different physical domains, you must use multi-domain blocks with ports representing different domains. The Single-Acting Hydraulic Cylinder block, shown in the figure, is one such block. A Hydraulic conserving port connects to a Hydraulic network, shown in yellow. A Mechanical Translational conserving port connects to a Mechanical Translational network, shown in green.

Mechanical Translational Reference   Single-Acting Hydraulic Cylinder   Mass

3-Way Directional Valve
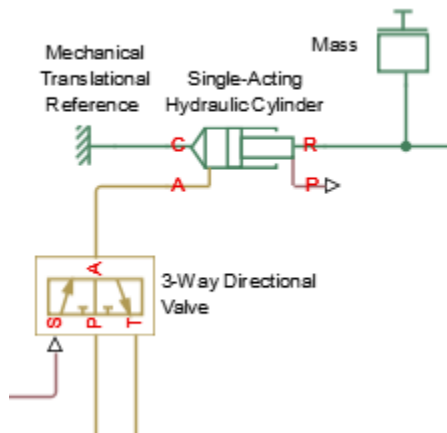
Physical signal ports similarly connect only to other physical signal ports. Physical signals can arise from blocks representing physical components, such as 2-Position Valve

Actuator, from Simscape sources, such as PS Constant, or from Simulink signals that have been converted into physical signals. The expected signal dimensions must be the same for all connected ports.

Signal Conversions

You must convert Simulink signals into physical signals if you intend to use them as Simscape block inputs. You perform this conversion using the Simscape `Simulink-PS Converter` block. The figure shows an example of such a conversion. A Simulink `Sine Wave` block generates a signal that the Simulink-PS Converter block converts for use in the Simscape Fluids `2-Position Valve Actuator` block.
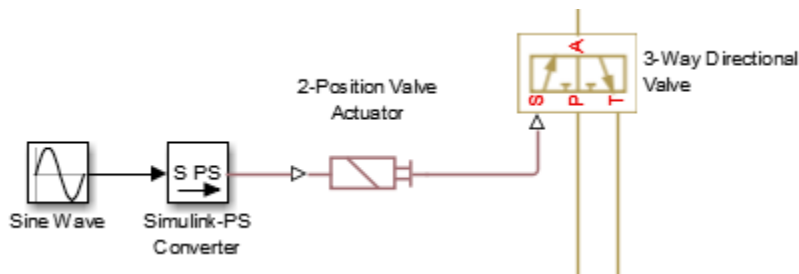
You must convert physical signals into Simulink signals if you intend to use them as Simulink block inputs. You perform this conversion using the Simscape `PS-Simulink Converter` block. The figure shows an example of such a conversion. A Simscape Fluids `Single-Acting Hydraulic Cylinder` block generates a signal that the PS-Simulink Converter block converts for plotting with a Simulink `Scope` block.
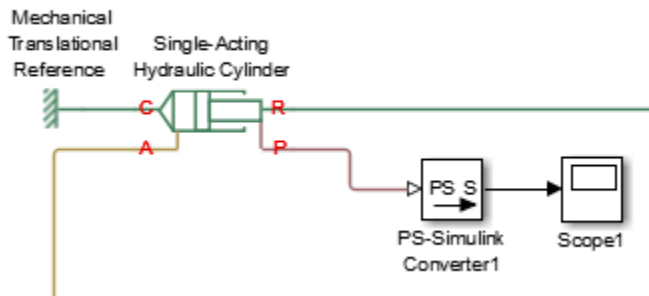
Model Equations

Simscape Fluids models are at their core systems of differential and algebraic equations that a numerical solver computes in terms of unknown variables. The equations come from blocks and domain definition files. Simscape software assembles the system of equations during model compilation and solves them for each time step during simulation.

Block equations are component-specific. They are often statements of mass, momentum, and energy conservation but can include equations such as heat transfer correlations and component geometry expressions. Domain equations apply model-wide and are based on general circuit analysis principles, often referred to as Kirchhoff's rules. These rules constrain the Across and Through variables so that:

- The Across variables of all directly connected conserving ports always have the same value.

$$P_1 = P_2 = P_3$$
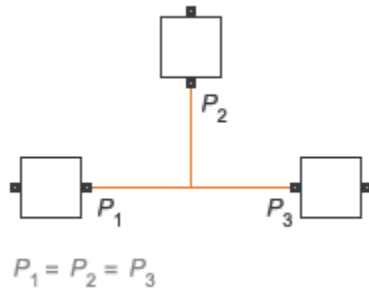
- The Through variables at a junction between two or more connection lines always sum to zero.

$$\dot{m}_1 + \dot{m}_2 + \dot{m}_3 = 0$$
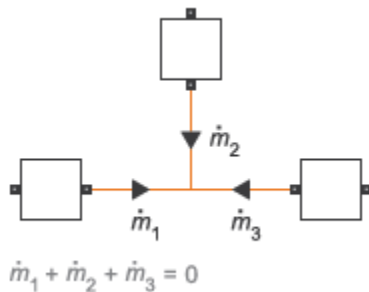
For more information on how Simscape software assembles and solves the governing equations of a model, see "How Simscape Simulation Works"

Example Model

The figure shows a Simscape Fluids model of a simple hydraulic actuator. The model has two physical networks—one hydraulic, indicated in yellow, and one Mechanical Translational, indicated in green. The Hydraulic network models the hydraulic power system that drives the actuator. The Mechanical Translational network models a damped mass-spring oscillator that the actuator pushes against.
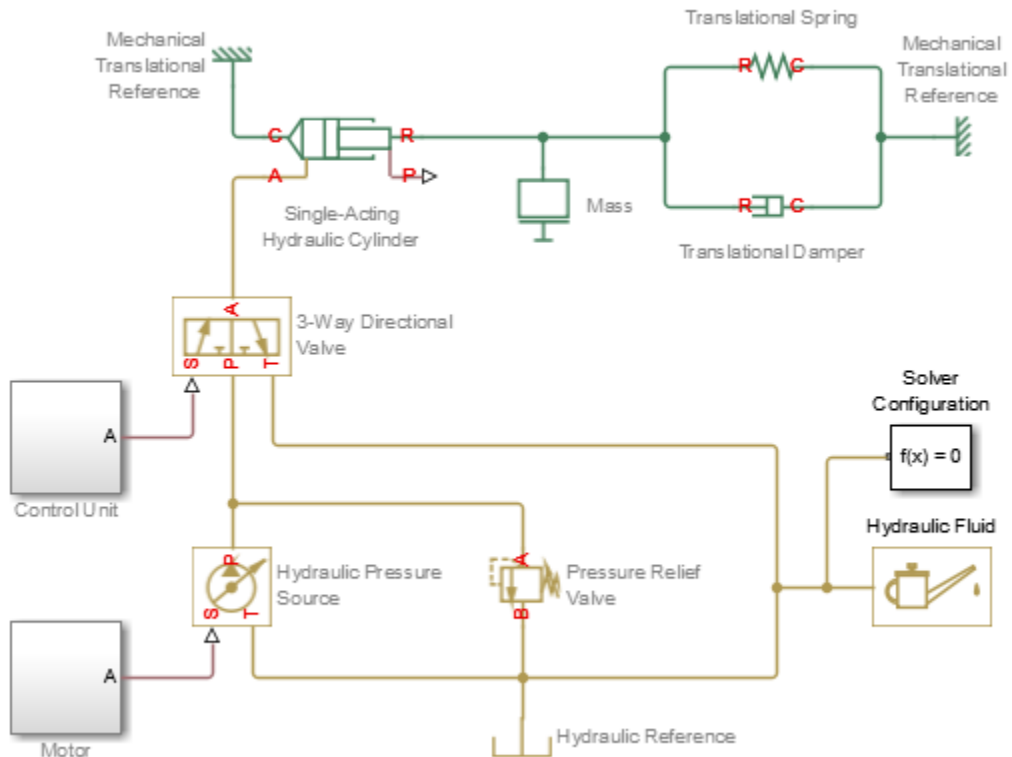
Hydraulic Actuator Simscape Fluids Model

The Hydraulic blocks represent the components of the hydraulic power system—a hydraulic pressure source, a directional valve, a relief valve, and the hydraulic fluid. The Mechanical Translational blocks represent the components of the damped mass-spring oscillator—a mass, a translational spring, and a translational damper.

The connection lines represent the physical connections between the components. For example, the connection line between the Hydraulic Pressure Source and 3-Way Directional Valve blocks indicates a direct connection between the actual components. This connection enables the flow of mass, energy, and momentum between the source and valve during simulation.

The physical connection lines join conserving ports of the same type. Hydraulic conserving ports connect to other Hydraulic conserving ports. Mechanical Translational conserving ports connect to other Mechanical Translational conserving ports. Hydraulic and Mechanical Translational components connect through a multi-domain component represented by the Single-Acting Hydraulic Cylinder block.

The physical signal lines represent Simscape block inputs. The physical signal line between the Control Unit subsystem and the 3-Way Directional Valve blocks specifies the valve position as a function of time. The subsystem block generates the position signal with Simulink blocks that a Simulink-PS Converter block then converts into a physical signal.

For more information about Simscape modeling rules, see "Basic Principles of Modeling Physical Networks". For a tutorial showing how to model an isothermal hydraulic actuator, see "Model an Isothermal Hydraulic Actuator" on page 2-33.

Specifying the Working Fluid

In this section...
"Specifying Fluid Properties" on page 2-10
"Selecting a Fluid Settings Block" on page 2-10
"Working with Multiple Fluid Networks" on page 2-10
"Omitting Fluid Properties in a Network" on page 2-11

Specifying Fluid Properties

Fluid properties impact your model dynamics and are required parameters for simulation. To specify them, select a fluid settings block from the appropriate physical domain and connect it anywhere on the target fluid network. The fluid settings block propagates the specified fluid properties to all components in the attached fluid network.

Selecting a Fluid Settings Block

- To set the fluid properties in a Thermal Liquid network, use the `Thermal Liquid Settings (TL)` block from the Simscape Foundation library. The block inputs are the tabulated fluid properties specified as functions of pressure and temperature.

- To set the fluid properties in a Hydraulic network, use the `Custom Hydraulic Fluid` block from the Simscape Foundation library. The block inputs are the fluid properties specified as constants.

  For an alternative way of specifying the fluid properties in a Hydraulic network, use the `Hydraulic Fluid` block from the Simscape Fluids Hydraulic (Isothermal Liquid) library. The block inputs include the fluid type—water, gasoline, jet fuel, and many others—selected from a dropdown list.

Working with Multiple Fluid Networks

Simscape Fluids models can have multiple fluid networks, each representing a single-fluid system. To avoid model compilation errors, use only one fluid settings block per network. Two fluid networks are treated as distinct if they belong to different Simscape domains or if they share no direct connection line.

Omitting Fluid Properties in a Network

If a fluid network has no fluid settings block, that network reverts to the default properties defined in the Simscape domain source files. The defaults correspond to water in the Thermal Liquid domain and to a typical hydraulic fluid in the Hydraulic domain.

- For the default thermal liquid properties, see "Thermal Liquid Domain".
- For the default hydraulic fluid properties, see "Hydraulic Domain".

Fluid Volumes in Thermal Liquid Networks

In this section...
"Fluid Volumes" on page 2-12
"Component Dynamics" on page 2-12
"Grounding Rules" on page 2-13
"Initial Conditions" on page 2-14
"Dynamic Components" on page 2-14

Fluid Volumes

Thermal Liquid components fall into two groups—those modeled with internal fluid volume and those not. A component has internal fluid volume if that volume is substantial and the internal fluid dynamics relevant. Components with fluid volume include pipes, tanks, and actuators. Components without fluid volume include valves, orifices, and sudden area changes. Fluid volumes enable you to:

- Model internal component dynamics.
- Satisfy the Simscape grounding rules.
- Specify initial conditions in a model.

Component Dynamics

Components with fluid volume have internal states, defined as differential variables, that can evolve over time. The internal states capture the transient effects of dynamic compressibility, fluid inertia, and wall compliance, when supported and enabled. State changes occur gradually, over the course of characteristic response times determined from component geometries and fluid properties. Such components are called *dynamic*.

Tank
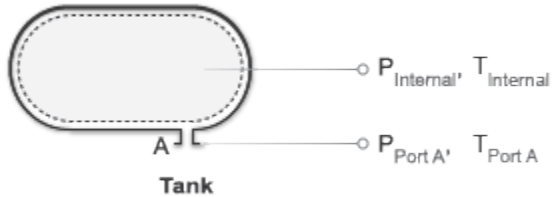
Dynamic Component

Components without fluid volume have no internal states. Algebraic equations constrain the states at the ports, but transient changes due to internal component dynamics are ignored. State changes occur instantaneously, without time lag. The components respond to dynamic changes just outside their ports, but so rapidly that they are, at each time step, at steady state with their neighbors. Such components are called *quasi-steady*.

Valve

Quasi-Steady Component

Grounding Rules

Dynamic components are required in Thermal Liquid models. These components have internal nodes at known pressure and temperature and therefore serve as state references in Thermal Liquid networks. You can satisfy the Simscape grounding rules by connecting at least one dynamic component to each Thermal Liquid network in your model.

Initial Conditions

To arrive at meaningful numerical solutions, the differential equations of dynamic components require initial conditions as inputs. The block dialog boxes of dynamic components provide initial condition parameters that you can set to change the initial configuration of your model. Quasi-steady components have no internal states and therefore no initial states to set.

The number of initial conditions should equal the number of degrees of freedom in your model. The number of degrees of freedom is the difference between the number of differential variables and the number of dynamic constraints. You can obtain both number using Simscape Statistics Viewer. You can view the initial conditions specified in a model using Simscape Variable Viewer.

Dynamic Components

The table lists the dynamic components that you can use, organized by library.

Library	Dynamic Components
Simscape Fluids > Thermal Liquid	`Single-Acting Actuator (TL)`
	`Double-Acting Actuator (TL)`
	`Pipe (TL)`
	`Tank (TL)`
	`Gas-Charged Accumulator (TL)`
Simscape Fluids > Fluid Network Interfaces	`E-NTU Heat Exchanger (TL-TL)`
	`E-NTU Heat Exchanger (TL)`
	`Heat Exchanger Interface (TL)`
Simscape > Foundation > Thermal Liquid	`Pipe (TL)`
	`Constant Volume Chamber (TL)`
	`Controlled Reservoir (TL)`
	`Reservoir (TL)`
	`Rotational Mechanical Converter (TL)`
	`Translational Mechanical Converter (TL)`

Energy Flows in Thermal Liquid Networks

In this section...

Upwind Energy Scheme

Thermal energy advection should always match the direction of flow. To satisfy this condition, the Thermal Liquid domain employs a modeling framework known as the upwind scheme. This scheme works by matching the specific internal energy at a port to that just upwind of the port:

- If the flow at a port is directed into the component, the specific internal energy at that port is equal to that carried by the incoming fluid. In the figure, the specific internal energy at port A of component **2** is equal to that at port B of component **1**.
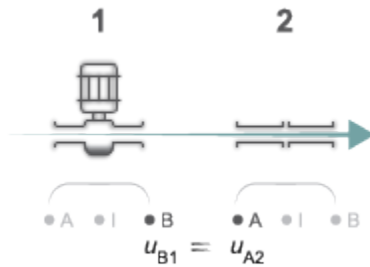
- If the fluid flow at a port is directed out of the component, the specific internal energy at that port is equal to that in the component's internal fluid volume. In the figure, the specific internal energy at port A of component **2** is equal to that in the internal fluid volume of the same component.

$$u_{A2} \doteq u_{I2}$$
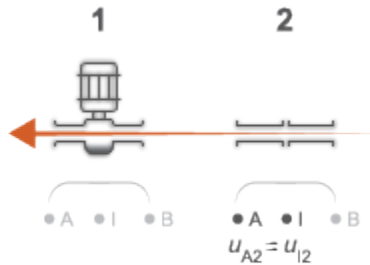
The upwind scheme describes the advection thermal energy flow rate using a conditional expression that, for port A of some component, reads:

$$\Phi_A^{Ad} = \begin{cases} \dot{m}_A u_A, & \dot{m}_A > 0 \\ \dot{m}_A u_I, & \dot{m}_A \le 0 \end{cases},$$

where:

- $\Phi_A{}^{Ad}$ is the thermal energy flow rate due to advection through port A.

- $\dot{m}_A$ is the mass flow rate through port A.

- u_A is the specific internal energy upwind of port A.

- u_I is the specific internal energy in the internal fluid volume.

Without numerical smoothing, this expression introduces a number of computational challenges. It adds a slope discontinuity to the thermal energy flux at zero mass flow rates that makes the model more prone to simulation errors. It also adds a jump discontinuity to the specific internal energy and makes its value ill-defined during flow reversals.
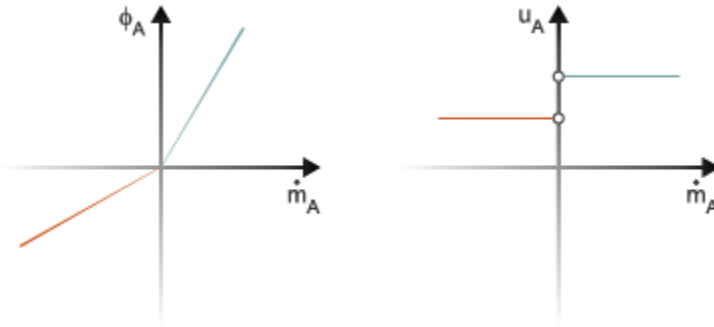
Upwind Energy Scheme without Smoothing

Numerical Smoothing

The Thermal Liquid domain smoothes the numerical discontinuities introduced by the upwind energy scheme by adding to the thermal energy flow rate a thermal conduction term. At port A of some component:

$$\Phi_A^{Th} = \begin{cases} \dot{m}_A u_A + G(u_A - u_I), & \dot{m}_A > 0 \\ \dot{m}_A u_I + G(u_A - u_I), & \dot{m}_A \leq 0 \end{cases},$$

where:

- Φ_A^{Th} is the thermal energy flow rate due to advection through port A and conduction between port A and the internal fluid volume.
- G is a thermal conductance coefficient computed from the fluid properties and component geometry.

Rewriting the modified expression in a more convenient form:

$$\Phi_A^{Th} = \begin{cases} (\dot{m}_A + G)u_A - Gu_I, & \dot{m}_A > 0 \\ Gu_A + (\dot{m}_A - G)u_I, & \dot{m}_A \leq 0 \end{cases}$$

Using `min` and `max` functions to collapse the conditional expression:

$$\Phi_A^{Th} = \left[\max(\dot{m}_A, 0) + G\right]u_A + \left[\min(\dot{m}_A, 0) - G\right]u_I$$

Approximating the collapsed expression as a numerically smooth expression:

$$\Phi_A^{Th} \approx \frac{\dot{m}_A + \sqrt{\dot{m}_A^2 + 4G^2}}{2}u_A + \frac{\dot{m}_A - \sqrt{\dot{m}_A^2 + 4G^2}}{2}u_I$$

The end expression removes the slope discontinuity from the thermal energy flow rate curve and the jump discontinuity from the specific internal energy curve. The thermal conductance coefficient determines the amount of smoothing applied in both cases:

$$G = \frac{kS}{c_v L},$$

where:

- k is the thermal conductivity defined in the Thermal Liquid Settings (TL) block.
- S is the flow cross-sectional area at the port considered.
- c_v is the specific heat defined in the Thermal Liquid Settings (TL) block.
- L is a characteristic distance between the port and the component's fluid volume.
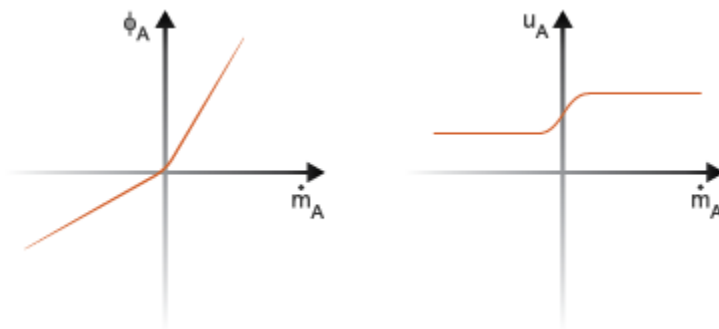
Upwind Energy Scheme with Smoothing

Total Energy Flow Rate

The Through variable of the Thermal Liquid domain is the total energy flow rate. This variable is defined in terms of the smoothed upwind thermal energy flow rate as:

$$\Phi_A = \Phi_A^{Th} + \frac{p}{\rho},$$

where:

- Φ_A is the total energy flow rate through port A.

- $\Phi_A{}^{Th}$ is the thermal energy flow rate through port A computed from the smoothed upwind energy scheme.

- p is the absolute pressure at port A.

- ρ is the fluid density at port A.

The kinetic energy contribution to the total energy flow rate is assumed negligible in Thermal Liquid networks and is ignored in this domain.

Model a Thermal Liquid Pipeline

Model Overview

This tutorial shows how to model a pump-driven thermal liquid pipeline. The pipeline system comprises a pump, a pipe, and two thermal liquid reservoirs. The pump provides power generation and the pipe fluid transport between the reservoirs. The reservoirs provide the pressure and temperature boundary conditions for the pipeline system.
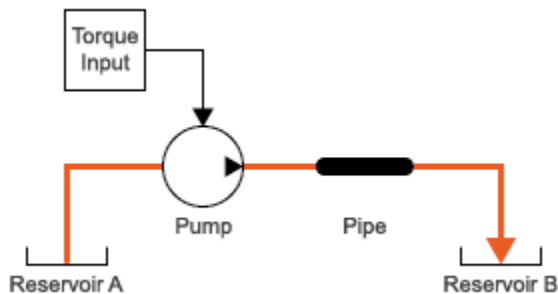
Thermal Liquid Pipeline Schematic

The reservoir conditions are fixed at room temperature and atmospheric pressure. The outer pipe surface temperature is fixed at a temperature of 275 K. The pipe fluid volume is initially at room temperature but, through the combined effects of viscous heating and conductive cooling, gradually shifts toward a new steady-state value determined by system dynamics.

The thermal resistance of the pipe is the result of convective and conductive individual resistances. A Pipe (TL) block models the convective thermal resistance at the inner pipe surface and the viscous heating in the thermal liquid. A Conductive Heat Transfer block models the conductive thermal resistance between the inner and outer pipe surfaces.
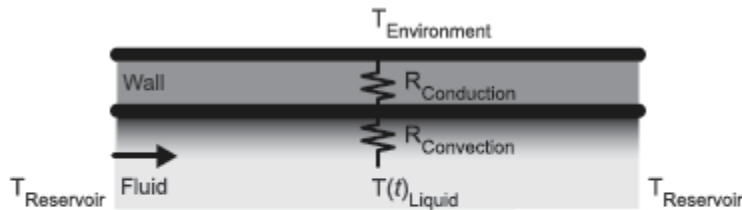
$$T_{Reservoir} = T(0)_{Liquid} = 293.15 \text{ K}$$
$$T_{Environment} = 275 \text{ K}$$

Pipe Thermal Resistances

Start a New Model

1 At the MATLAB command prompt, enter `ssc_new`. MATLAB opens the Simscape model template. This template provides a starting point for your Simscape Fluids model. Save the model often as you add new blocks.

2 From the **Simscape Foundation** > **Thermal Liquid** > **Utilities** library, drag a `Thermal Liquid Settings (TL)` block to the model canvas. This block defines the physical properties of the thermal liquid, including its viscosity, thermal conductivity, and bulk modulus.

3 Connect the `Solver Configuration` and Thermal Liquid Settings (TL) blocks as shown in the figure. The Solver Configuration block provides Simscape solver settings for your model. Each topologically distinct physical network requires one such block.

Model the Pipeline

1 From the **Simscape Foundation** > **Thermal Liquid** > **Elements** library, drag two `Reservoir (TL)` blocks. The reservoir blocks set the pressure and temperature boundary conditions for the pipeline model.

2 From the **Simscape Fluids** > **Thermal Liquid** library drag these blocks.

Block	Library
Pipe (TL)	**Pipes & Fittings**
Fixed-Displacement Pump (TL)	**Pumps & Motors**

The pipe block provides a conduit for thermal liquid transport between the reservoirs. The pump block provides a power source to generate a positive mass flow rate through the pipe.

3 Connect the blocks as shown in the figure. The left reservoir serves as a thermal liquid source. The right reservoir serves as a thermal liquid sink.
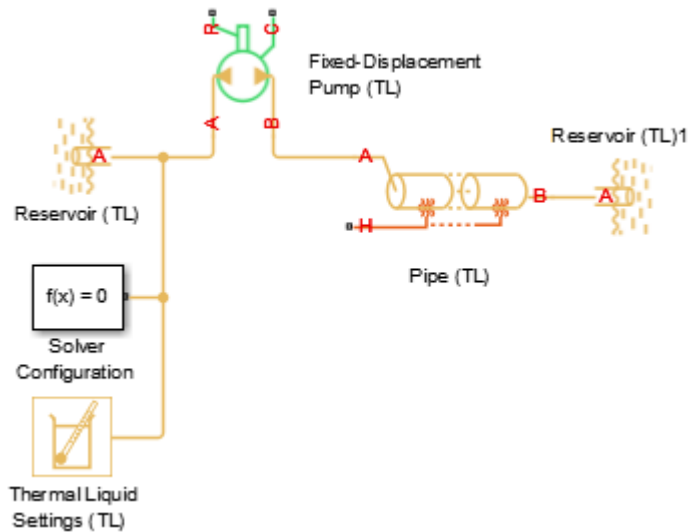
The component colors indicate the physical domains they represent—yellow for Thermal Liquid, orange for Thermal, and green for Mechanical Rotational. The pump and pipe blocks are multidomain blocks that you can use to interface different physical domains.

4 In the Pipe (TL) block, set the **Pipe length** parameter to 1000 m. This length makes more evident the thermal liquid heating due to viscous friction losses.

Model the Thermal Conduction

1 From the Simscape Foundation library, drag these blocks.

Block	Library
Conductive Heat Transfer	**Thermal** > **Thermal Elements**
Thermal Reference	**Thermal** > **Thermal Elements**
Ideal Temperature Source	**Thermal** > **Thermal Sources**

The heat transfer block models the conductive heat losses through the pipe wall to the surrounding environment. The ideal temperature source sets the environment

temperature through a physical signal input. The thermal reference block provides a reference against which to specify the temperature physical signal input.

2 From the **Simscape** > **Foundation** library, drag a `PS Constant` block. This block enables you to specify the numerical value of the environment temperature.

3 Connect the blocks as shown in the figure. The new blocks representing the conductive heat losses belong to the Simscape thermal domain and connect only to thermal conserving ports.
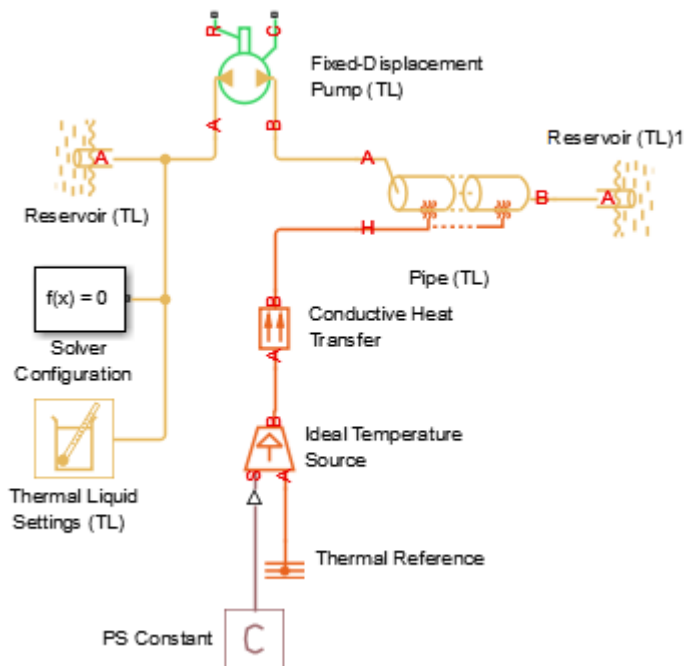
4 In the PS Constant block dialog box, set the **Constant** parameter to `275`. This value is the environment temperature in the default temperature units of Kelvin.

5 In the Conductive Heat Transfer block, set the **Area** parameter to `pi*0.1128*1000`. This expression gives the pipe surface area in terms of the pipe hydraulic diameter (`0.1128` m) and length (`1000` m). The default **Thermal conductivity** value is that of copper, a highly conductive metal.

Model the Pump Torque

1 From the Simscape Foundation library, drag these blocks.

Block	Library
Ideal Torque Source	**Mechanical** > **Rotational Sources**
Mechanical Rotational Reference	**Mechanical** > **Rotational Elements**

The ideal torque source block sets the pump driving torque through a physical signal input. The Mechanical Rotational reference block provides a reference against which to specify the torque physical signal input.

2 From the **Simscape** > **Utilities** library, drag a `PS Constant` block. This block specifies the numerical value of the pump torque.

3 Connect the blocks as shown in the figure. The new blocks representing the pump power source belong to the Mechanical Rotational domain and connect only to Mechanical Rotational conserving ports.
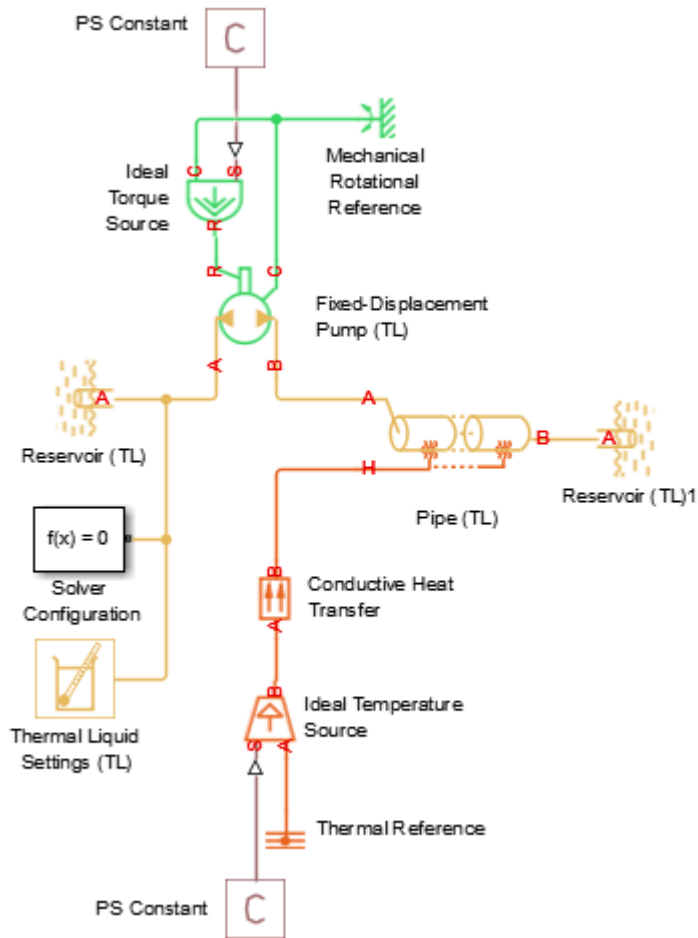
Port R of the pump block represents the rotating shaft and connects to the ideal torque source block. Port C represents the pump casing and connects to the mechanical rotational reference.

4   In the PS Constant block dialog box, set the **Constant** parameter to 50. This value specifies the pump torque in the default torque units of N*m.

Add Thermal Liquid Sensors

1 Drag these blocks to the model.

Block	Library
Mass & Energy Flow Rate Sensor (TL)	**Simscape Foundation** > **Thermal Liquid** > **Sensors**
PS-Simulink Converter	**Simscape** > **Utilities**
Scope	**Simulink** > **Sinks**

The sensor block measures the Through variables of the Thermal Liquid domain and outputs them as physical signals. The converter block transforms the physical signal into a Simulink signal for plotting in a Scope block.

2 Connect the blocks as shown in the figure. Arrange the sensor block so that both of its conserving ports connect to the Thermal Liquid branch being probed. The sensor output is the measured flow rate in the direction of port A to port B.
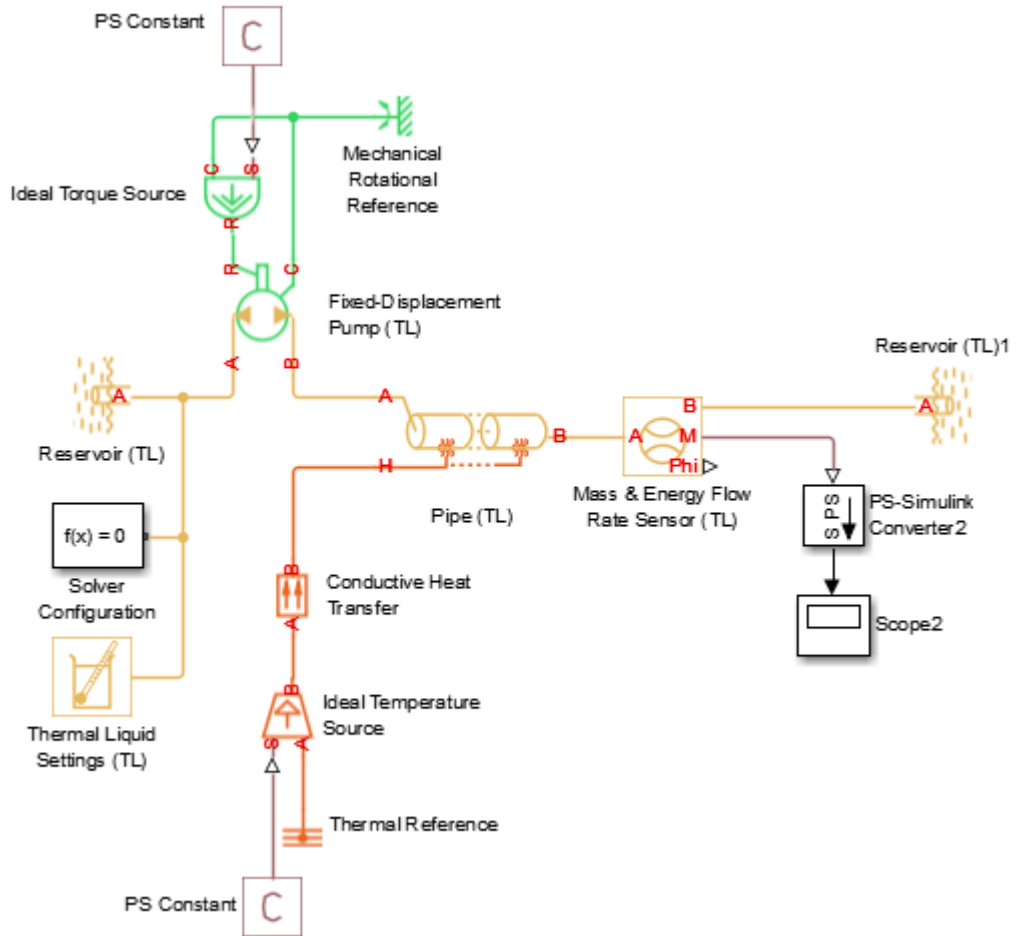
3 Drag these blocks to the model.

Block	Library
Pressure & Temperature Sensor (TL)	**Simscape Foundation > Thermal Liquid > Sensors**
Absolute Reference (TL)	**Simscape Foundation > Thermal Liquid > Elements**

Block	Library
PS-Simulink Converter	**Simscape** > **Utilities**
Scope	**Simulink** > **Sinks**

The sensor block measures the Across variables of the Thermal Liquid domain and outputs them as physical signals. The converter block transforms the physical signal into a Simulink signal for plotting in a Scope block.

4   Connect the blocks as shown in the figure. Arrange the sensor block so that one conserving port connects to the Thermal Liquid branch being probed and the other to a known pressure and temperature reference, such as that provided by an Absolute Reference (TL) block. The sensor output is the difference between the measured and reference quantities, taken from port A to port B.
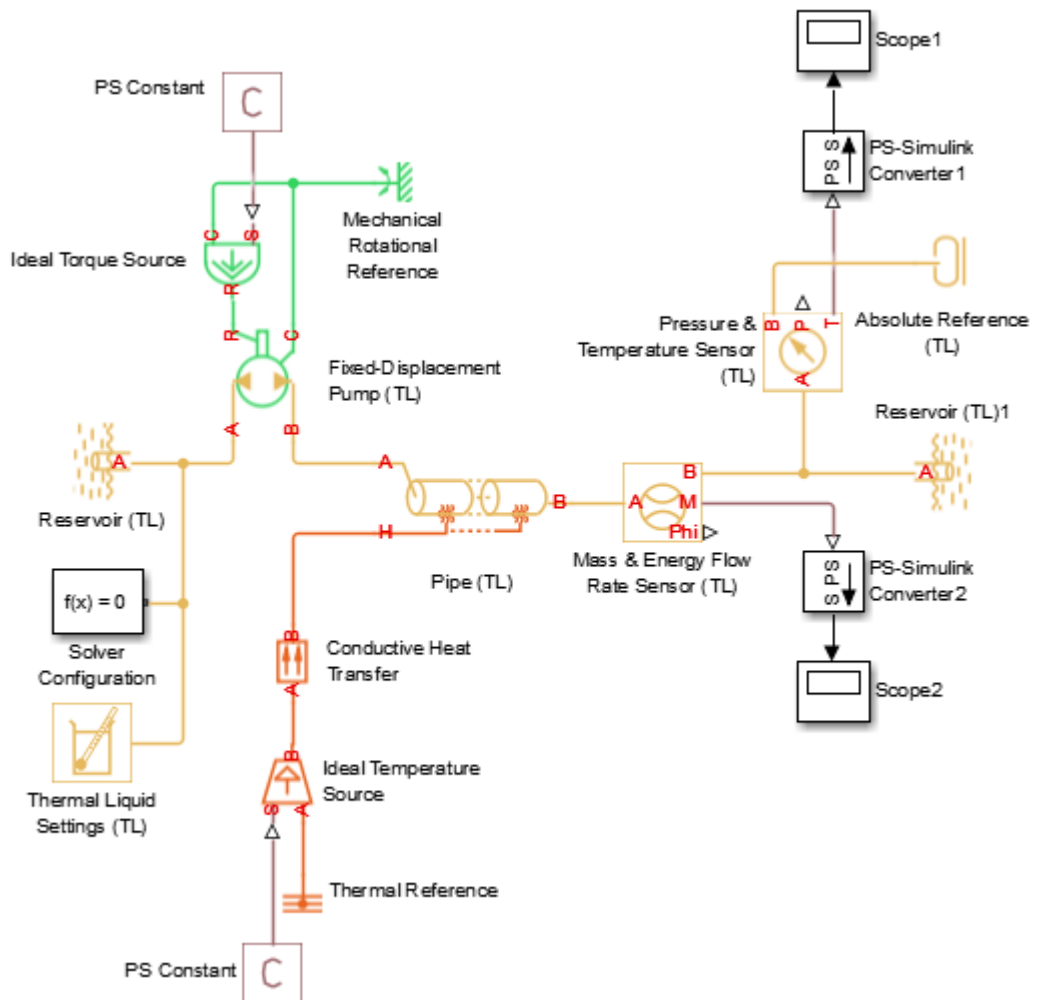
Simulate the Pipeline Model

The Simscape model template specifies suitable solver settings for this model. Run the simulation, for example, by clicking the ▶ button in the Simulink toolstrip. Open the

mass flow rate Scope block. The plot shows a gradual change in flow rate as the thermal liquid transitions from a state of rest to a new steady-state velocity.
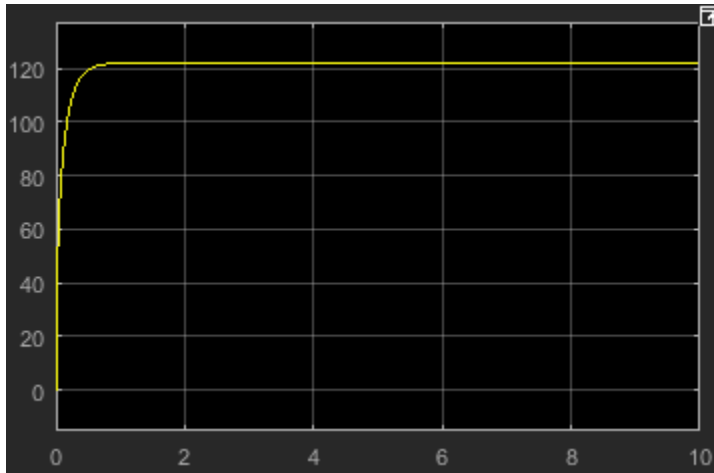
The time scale for temperature changes is substantially greater than the 10-second simulation time. To see the new steady-state temperature, change the simulation stop time to 200 s. Then, simulate the model and open the temperature Scope block. The plot shows the gradual cooling at the pipe outlet as the thermal liquid transitions to a new steady-state temperature.
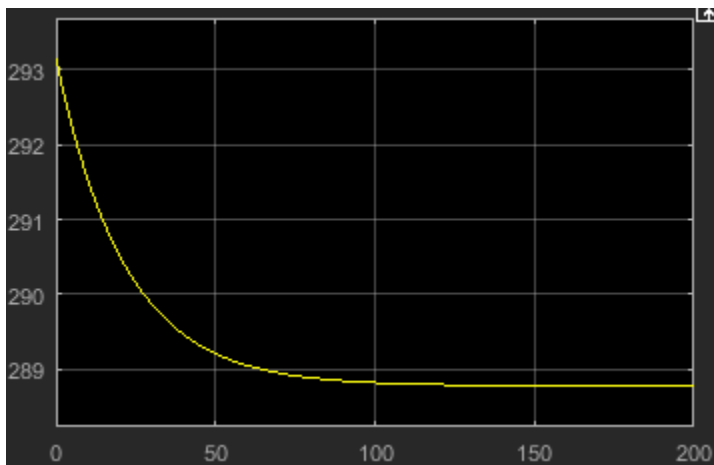
Experiment with the block settings and input signals. Try setting the Pipe (TL) block **Fluid inertia** parameter to On. The figure shows the combined effects of fluid dynamic compressibility and fluid inertia on the mass flow rate plot. The simulation stop time is 10 seconds for this plot.
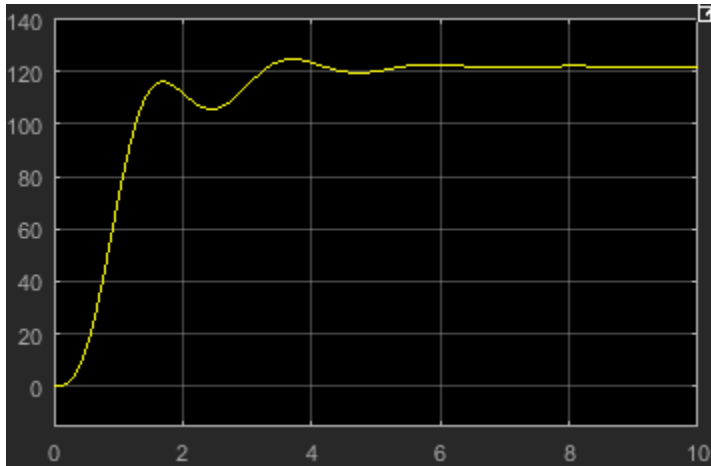
Model an Isothermal Hydraulic Actuator

In this section...
"Model Overview" on page 2-33
"Start a New Model" on page 2-35
"Model the Actuator" on page 2-36
"Model the Actuator Load" on page 2-38
"Add Sources and Scopes" on page 2-39
"Simulate the Actuator Model" on page 2-41

Model Overview

This tutorial shows how to model an isothermal hydraulic actuator using Simscape Fluids blocks. The actuator comprises a pump, a valve, and a cylinder. The pump provides power generation, the valve flow regulation, and the cylinder power conversion into mechanical translational form. A lumped mass-spring-damper system provides a simple model of the actuator load.

The figure shows a schematic of the complete actuator system. The model makes some simplifying assumptions. An ideal pressure source replaces the pump as a power source. Simple physical signal inputs replace the motor component and the control unit algorithm. The relief valve is unnecessary detail in this example and is ignored.
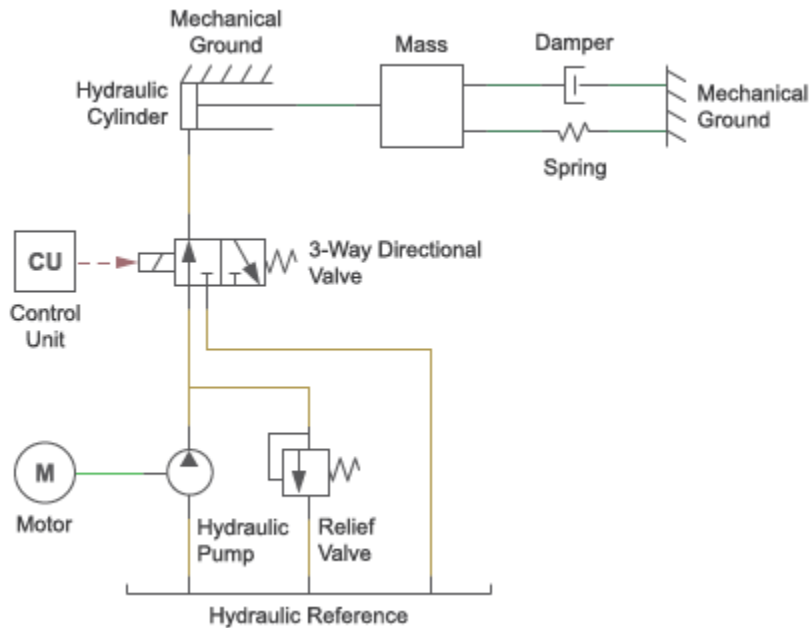
Hydraulic Actuator Schematic

The 3-way directional valve provides three ports and two control positions. The ports represent the cylinder, pump, and hydraulic reference inlets. The control position varies with the valve actuator input. Above 50% of the valve actuator nominal signal value, the valve opens the pump-cylinder flow path. The cylinder shaft extends against the actuator load. Below 50% of the valve actuator nominal signal value, the valve opens the cylinder-reservoir flow path. The cylinder shaft retracts under the action of the actuator load.
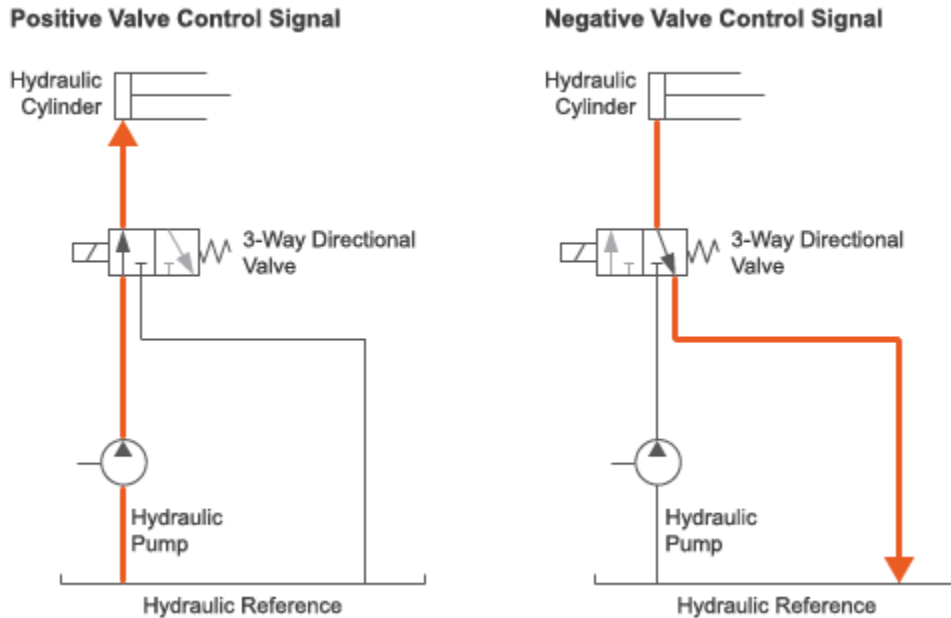
Directional Valve Positions

Start a New Model

1 At the MATLAB command prompt, enter ssc_new. MATLAB opens the Simscape model template. This template provides a starting point for your Simscape Fluids model. Save the model often as you add new blocks.

2 From the **Hydraulics (Isothermal)** > **Hydraulic Utilities** library, drag a Hydraulic Fluid block to the model canvas. This block defines the physical properties of the hydraulic fluid, including its density, viscosity, and bulk modulus.

3 Connect the Solver Configuration and Hydraulic Fluid blocks as shown in the figure. The Solver Configuration block provides Simscape solver settings for your model. Each topologically distinct physical network requires one such block.
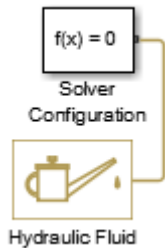
Model the Actuator

1   From the **Simscape** > **Foundation** library, drag these blocks to the model canvas.

Block	Sublibrary
Hydraulic Constant Pressure Source	**Hydraulic** > **Hydraulic Sources**
Hydraulic Reference	**Hydraulic** > **Hydraulic Elements**

The pressure source represents the pump. The hydraulic reference represents the reservoir that the pump draws fluid from.

2   From the **Simscape** > **Fluids** library, drag these blocks.

Block	Sublibrary
Single-Acting Hydraulic Cylinder	**Hydraulics (Isothermal)** > **Hydraulic Cylinders**
3-Way Directional Valve	**Hydraulics (Isothermal)** > **Valves**
2-Position Valve Actuator	**Hydraulics (Isothermal)** > **Valves**

The hydraulic cylinder converts the hydraulic power generated by the pressure source into a mechanical translational form. The directional valve and valve actuator regulate the flow to the cylinder. In so doing, they control the mechanical power generated at the cylinder.

3   Connect the blocks as shown in the figure. The 3-Way Directional Valve and Hydraulic Constant Pressure Source blocks must both connect to the Hydraulic Reference block.
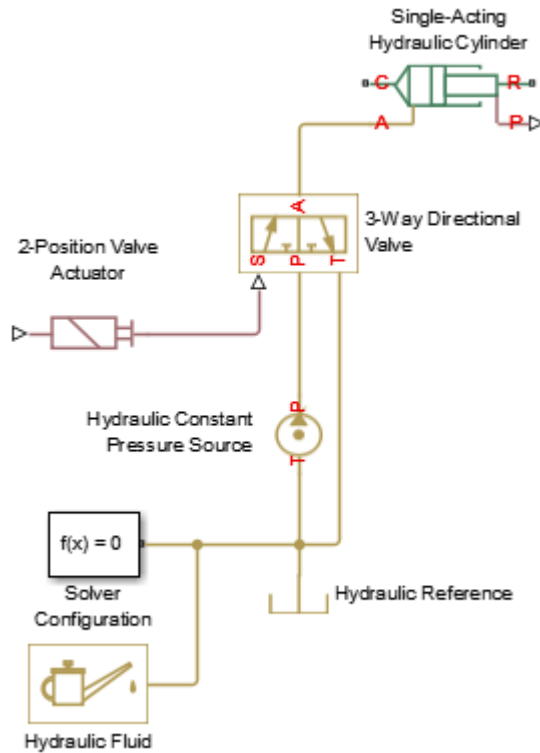
The block connections mirror the physical connections between the actual components. The cylinder connects to the valve, which connects to the pump, which in turn connects to the fluid reservoir.

4 In the 3-Way Directional Valve block dialog box, set the **Orifice A-T initial opening** parameter to `0.01`.

5 From the Simscape Multibody Foundation library, drag a `Mechanical Translational Reference` block. This block appears in the **Mechanical** > **Translational Elements** sublibrary.

6 Connect the Mechanical Translational Reference block to the Single-Acting Hydraulic Cylinder block as shown in the figure.
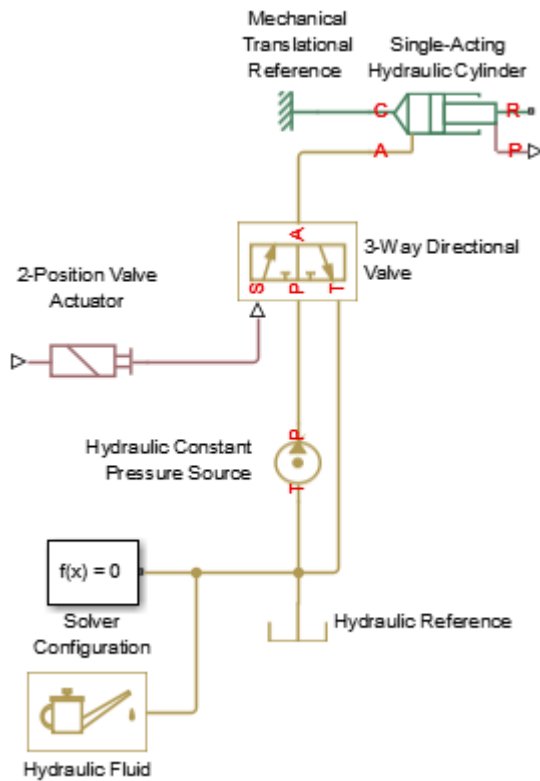
The connection line fixes the hydraulic cylinder casing, represented by port c, to a mechanical ground so that only the shaft, represented by port R, is capable of motion.

Model the Actuator Load

1   From the **Simscape** > **Foundation** > **Mechanical** > **Translational Elements** library, drag these blocks to the model canvas:

   · Mass

   · Translational Spring

   · Translational Damper

- `Mechanical Translational Reference`

2 Connect the blocks as shown in the figure.
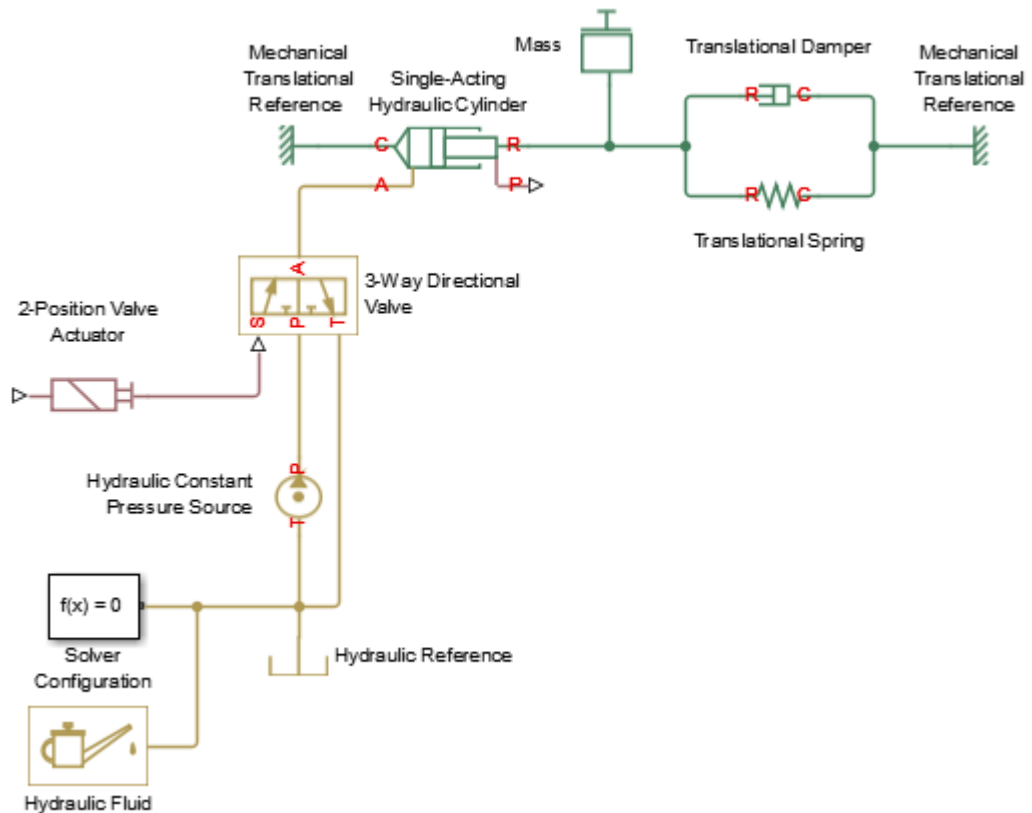
The mass, spring, and damper elements provide a simple model of the actuator load. The mass accounts for kinetic energy storage, the spring for potential energy storage, and the damper for viscous energy dissipation due to motion.

Add Sources and Scopes

1 From the Simulink Sources library, drag a `Sine Wave` and two `Scope` blocks.

2 From the Simscape Utilities library, drag one `Simulink-PS Converter` and two `PS-Simulink Converter` blocks.

3 Connect the Sine Wave, Simulink-PS Converter, and 2-Position Valve Actuator blocks in series as shown in the figure.
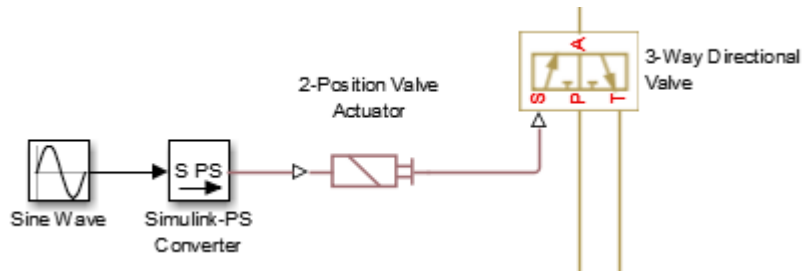
The Sine Wave block provides a sinusoidal control input to the 2-Position Valve Actuator block. The Simulink-PS Converter block converts the sinusoidal input from a Simulink signal into a physical signal compatible with Simscape Multibody blocks.

4 In the Sine Wave block dialog box, set the **Amplitude parameter** to 24. This value corresponds to the default nominal signal value of the 2-Position Valve Actuator block.

5 Connect the Single-Acting Hydraulic Cylinder block, a PS-Simulink Converter block, and a Scope block in series as shown in the figure.
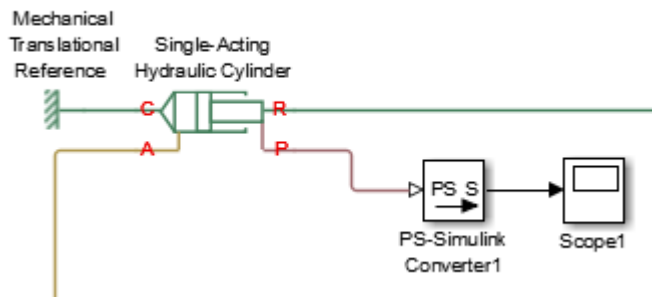
Port p of the Single-Acting Hydraulic Cylinder block outputs a physical signal with the measured cylinder shaft position. The PS-Simulink Converter block converts the physical signal into a Simulink signal for plotting with the Simulink Scope block.

6 Connect the output side of the 2-Position Valve Actuator block, a PS-Simulink Converter block, and a Scope block in series as shown in the figure.
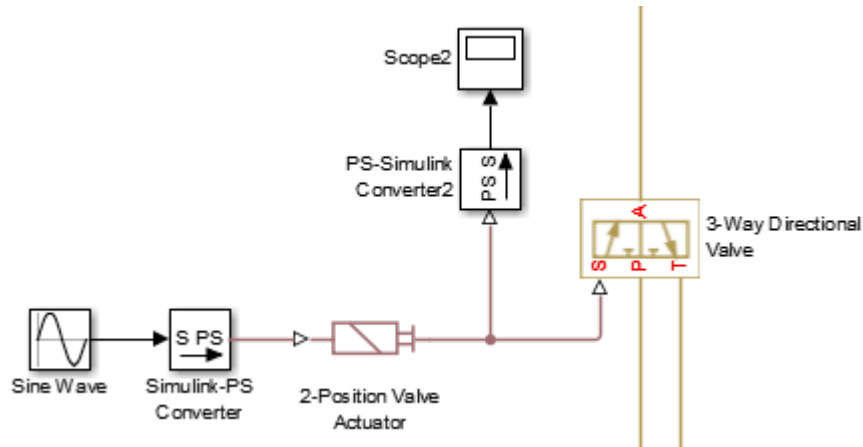
The output side of the 2-Position Valve Actuator block outputs a physical signal with the measured valve opening. The PS-Simulink Converter block converts the physical signal into a Simulink signal for plotting with the Simulink Scope block.

Simulate the Actuator Model

1 Run the simulation. You can do this from the Simulink menu bar, by clicking

the ⊙ button. Simscape software evaluates the model, determines the initial conditions, and simulates the model. For more information, see "How Simscape Simulation Works".

2 Open the Scope blocks.

The valve opening plot shows the dynamic response of the valve to the sinusoidal wave input signal.
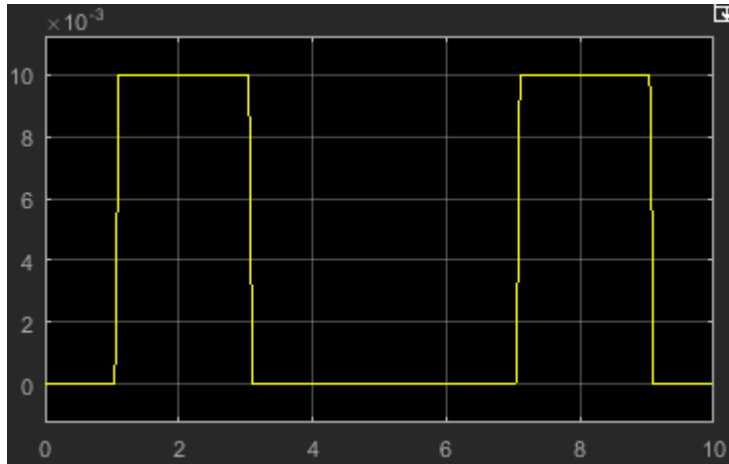
The cylinder position plot shows the dynamic response of the cylinder shaft to the pressure changes in the cylinder chamber.
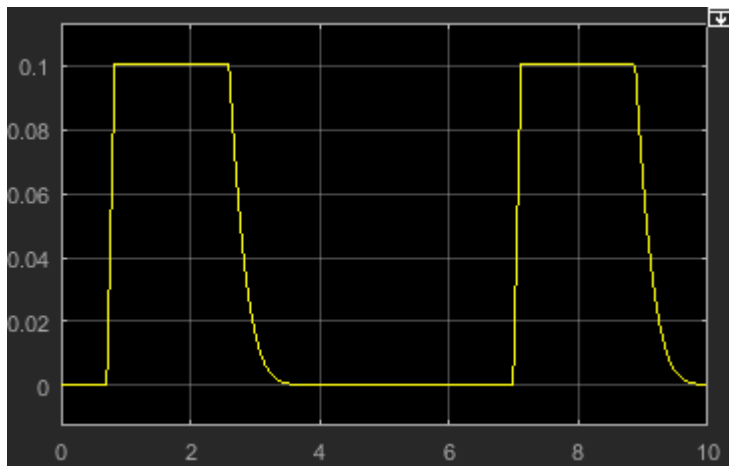
The valve is initially closed. When the valve control signal reaches 50% of the actuator nominal signal, the valve gradually opens to allow flow between the pump and the cylinder. The cylinder shaft moves in the positive direction until it reaches the maximum position allowed by the piston stroke.

When the valve signal goes below 50% of the actuator nominal signal, the valve gradually closes to prevent flow from the pump to the cylinder. The spring element in the actuator load pushes back against the cylinder shaft, forcing it to return to its retracted position.

Experiment with different valve inputs and block parameter values. Simulate the model under different conditions to see the effect of different inputs and parameters on the valve opening dynamics and cylinder shaft displacement.

Modeling Power Units

The power unit is perhaps the most prevalent unit in hydraulic systems. Its main function is to supply the required amount of fluid under specified pressure. There is a wide variety of power unit designs varying by the amount and type of pumps, prime movers, valves, tanks, etc. The set of blocks available in the Simscape Fluids libraries allows you to simulate practically any of these configurations. This section considers basic approaches in simulating power units and examples of typical schematics.

A typical power unit of a hydraulic system, as shown in the following illustration, consists of a fixed-displacement or variable-displacement pump, reservoir, pressure-relief valve, and a prime mover that drives the hydraulic pump.
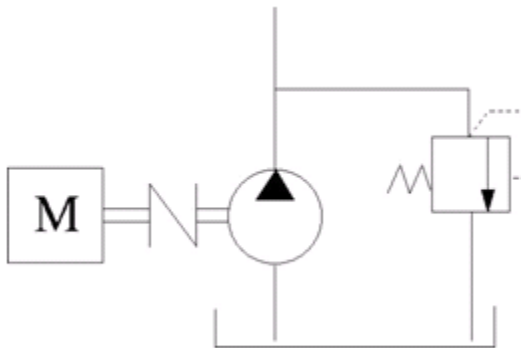
Typical Hydraulic Power Unit

In developing a model of a power unit, you must reach a compromise between the robustness, speed of simulation, and accuracy, meaning that the model should be as simple as possible to provide acceptable accuracy within the working range of variable parameters.

The first option is to simulate a power unit literally, as it is, reproducing all its components. This approach is illustrated in the Closed-Loop Pump Control with Flexible Drive Shaft example. The power unit consists of a fixed-displacement pump, which is driven by a motor through a compliant transmission, a pressure-relief valve, and a variable orifice, which simulates system fluid consumption. The motor model is represented as a source of angular velocity rotating shaft at 188 rad/s at zero torque.

The load on the shaft decreases the velocity with a slip coefficient of 1.2 (rad/s)/Nm. The load on the driving shaft is measured with the torque sensor. The shaft between the motor and the pump is assumed to be compliant and simulated with rotational spring and damper.

The simulation starts with the variable orifice opened, which results in a low system pressure and the maximum flow rate going to the system. The orifice starts closing at 0.5 s, and is closed completely at 3 s. The output pressure builds up until it reaches the pressure setting of the relief valve (75e5 Pa) and is maintained at this level by the valve. At 3 s, the variable orifice starts opening, thus returning system to its initial state.

You can implement a considerably more complex model of a prime mover by following the pattern used in the example. For instance, the shaft can be represented with multiple segments and intermediate bearings. The model of a prime mover can be more comprehensive, accounting for its type (DC or AC electric motor, diesel or gasoline engine), characteristics, control type, and so on. In addition, a complex mechanical transmission driven by a diesel or gasoline internal combustion engine modeled using Simscape Driveline software can be combined with the Simscape Fluids model of the hydraulic portion of a power unit.

Depending on your particular application, you may be able to simplify the model of a power unit practically without a loss in accuracy. The main factors to be considered in this process are the driving shaft angular velocity variation magnitude and the system pressure variation range. If the prime mover angular velocity remains practically constant during simulated time or varies insignificantly with respect to its steady-state value, the entire driving shaft subsystem can be replaced with the Ideal Angular Velocity Source block, whose output is set to the steady-state value, as it is shown in the following illustration.
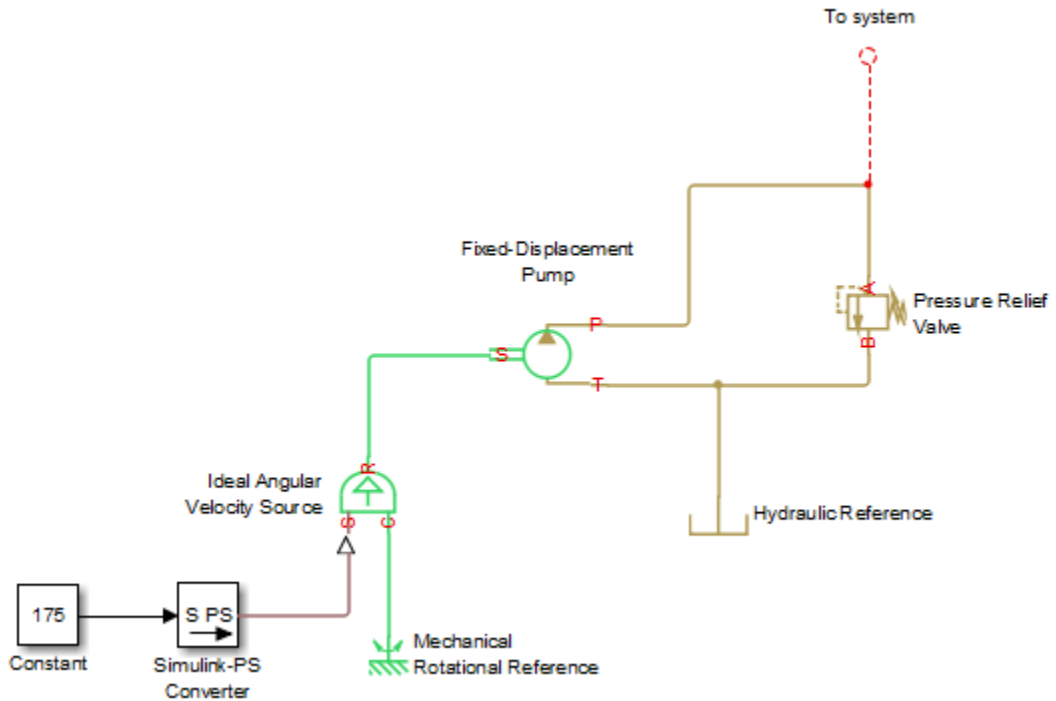
Using the Ideal Angular Velocity Source Block in Modeling Power Units

Furthermore, if pump delivery exceeds the system's fluid requirements at all times, the pump output pressure remains practically constant and close to the pressure setting of the pressure-relief valve. If this assumption is true and acceptable, the entire power unit can be reduced to an ideal Hydraulic Pressure Source block, as shown in the next illustration.
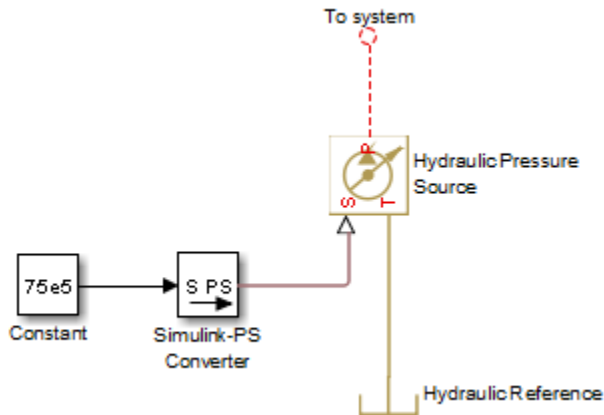
Using the Hydraulic Pressure Source Block in Modeling Power Units

The two previous examples demonstrate that the use of ideal sources is a powerful means of reducing the complexity of models. However, you should exercise extreme caution every time you use an ideal source instead of a real pump. The substitution is possible only if there is an assurance that the controlled parameter (angular velocity in the first example, and pressure in the second example) remains constant. If this is not the case, the power unit represented with an ideal source will generate considerably more power than its simulated physical counterpart, thus making the simulation results incorrect.

Modeling Directional Valves

In this section...

Types of Directional Valves

The main function of directional valves in hydraulic systems is to direct and distribute flow between consumers. As far as the valve modeling is concerned, the valves are classified by the following main characteristics:

- Number of external paths (connecting ports) — One-way, two-way, three-way, four-way, multiple-way
- Number of positions a control member of the valve can assume — Two-position, three-position, multiple-position, continuous (can assume any position within working range)
- Control member type — Spool, poppet, sliding flat spool, and so on

As an example, the following illustration shows a portion of a hydraulic system with a 4-way, 3-position directional valve controlling a double-acting cylinder, next to its schematic diagram.
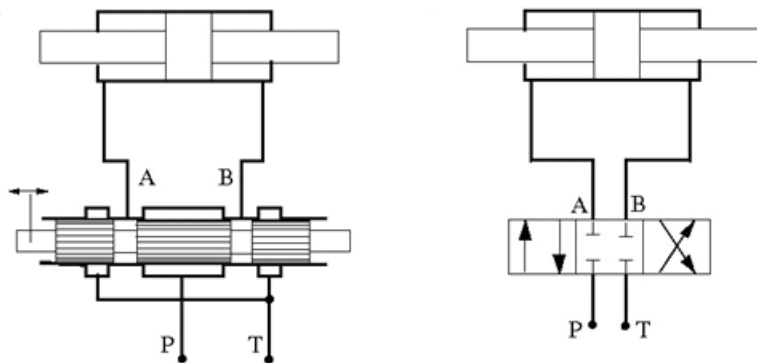
Throughout Simscape Fluids libraries, Hydraulic (isothermal liquid) ports are identified with the following symbols:
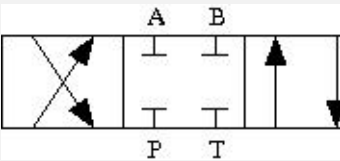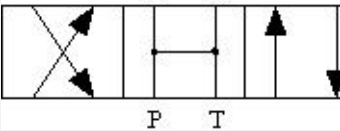
- P — Pressure port
- T — Return (tank) port
- A, B — Actuator ports
- X, Y — Pilot or control ports
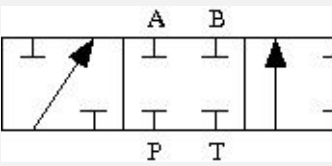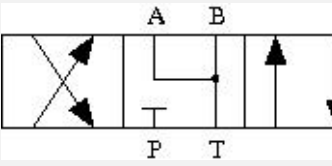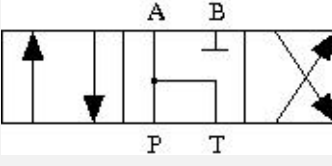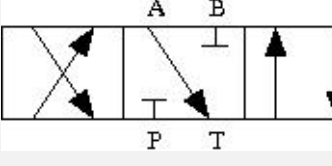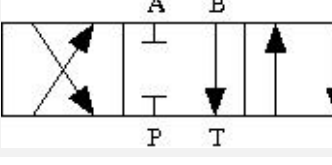
4-Way Directional Valve Configurations
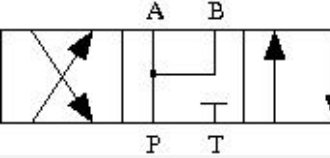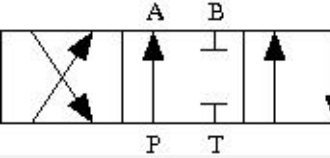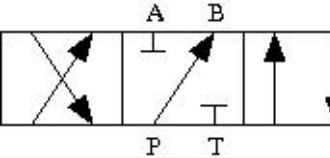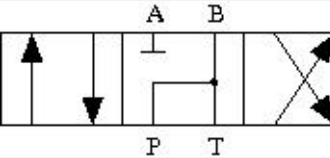
4-way directional valves are available in multiple configurations, depending on the port connections in three distinctive valve positions: leftmost, neutral, and rightmost. Each configuration is characterized by the number of variable orifices, the way the orifices are connected, and initial openings of the orifices. Ten 4-way directional valve blocks in Simscape Fluids libraries represent twenty most typical valve configurations. Configurations that differ only by the values of initial openings are covered by the same model.

The basic `4-Way Directional Valve` block lets you model eleven most popular configurations by changing the initial openings of the orifices, as shown in the following table.

Basic 4-Way Directional Valve Configurations

No	Configuration	Initial Openings
1		All four orifices are overlapped in neutral position: • **Orifice P-A initial opening** < 0 • **Orifice P-B initial opening** < 0 • **Orifice A-T initial opening** < 0 • **Orifice B-T initial opening** < 0
2		All four orifices are open (underlapped) in neutral position: • **Orifice P-A initial opening** > 0 • **Orifice P-B initial opening** > 0 • **Orifice A-T initial opening** > 0 • **Orifice B-T initial opening** > 0

No	Configuration	Initial Openings
3		Orifices P-A and P-B are overlapped. Orifices A-T and B-T are overlapped for more than valve stroke: • **Orifice P-A initial opening** < 0 • **Orifice P-B initial opening** < 0 • **Orifice A-T initial opening** $< -valve\_stroke$ • **Orifice B-T initial opening** $< -valve\_stroke$
4		Orifices P-A and P-B are overlapped, while orifices A-T and B-T are open: • **Orifice P-A initial opening** < 0 • **Orifice P-B initial opening** < 0 • **Orifice A-T initial opening** > 0 • **Orifice B-T initial opening** > 0
5		Orifices P-A and A-T are open in neutral position, while orifices P-B and B-T are overlapped: • **Orifice P-A initial opening** > 0 • **Orifice P-B initial opening** < 0 • **Orifice A-T initial opening** > 0 • **Orifice B-T initial opening** < 0
6		Orifice A-T is initially open, while all three remaining orifices are overlapped: • **Orifice P-A initial opening** < 0 • **Orifice P-B initial opening** < 0 • **Orifice A-T initial opening** > 0 • **Orifice B-T initial opening** < 0
7		Orifice B-T is initially open, while all three remaining orifices are overlapped: • **Orifice P-A initial opening** < 0 • **Orifice P-B initial opening** < 0

No	Configuration	Initial Openings
		• **Orifice A-T initial opening** < 0
		• **Orifice B-T initial opening** > 0
8		Orifices P-A and P-B are open, while orifices A-T and B-T are overlapped: • **Orifice P-A initial opening** > 0 • **Orifice P-B initial opening** > 0 • **Orifice A-T initial opening** < 0 • **Orifice B-T initial opening** < 0
9		Orifice P-A is initially open, while all three remaining orifices are overlapped: • **Orifice P-A initial opening** > 0 • **Orifice P-B initial opening** < 0 • **Orifice A-T initial opening** < 0 • **Orifice B-T initial opening** < 0
10		Orifice P-B is initially open, while all three remaining orifices are overlapped: • **Orifice P-A initial opening** < 0 • **Orifice P-B initial opening** > 0 • **Orifice A-T initial opening** < 0 • **Orifice B-T initial opening** < 0
11		Orifices P-B and B-T are open, while orifices P-A and A-T are overlapped: • **Orifice P-A initial opening** < 0 • **Orifice P-B initial opening** > 0 • **Orifice A-T initial opening** < 0 • **Orifice B-T initial opening** > 0

The other nine configurations are covered by the remaining 4-way directional valve blocks (A through K), as shown in the next table.

Other 4-Way Directional Valve Blocks

Block Name	Configuration	Description
4-Way Directional Valve A		Contains two additional normally-open, sequentially-located orifices. Valve displacement to the left or to the right closes the path to tank.
4-Way Directional Valve B		Ports P and A are permanently connected through fixed orifice. Orifices P-B and B-T are initially open (underlapped).
4-Way Directional Valve C		Ports P and B are permanently connected through fixed orifice. Orifices P-A, A-T and B-T are initially open (underlapped).
4-Way Directional Valve D		Two orifices are installed in the P-A link. Port A never connects to port T.
4-Way Directional Valve E		Two orifices are installed in the P-B link. Port B never connects to port T.
4-Way Directional Valve F		Two parallel orifices in the P-A arm and two series orifices in the A-T arm.

Block Name	Configuration	Description
4-Way Directional Valve G		Two parallel orifices in the P-B arm and two series orifices in the B-T arm.
4-Way Directional Valve H		Two parallel orifices in the P-B arm and two series orifices in the P-T arm.
4-Way Directional Valve K		Two parallel orifices in the P-A arm and two series orifices in the P-T arm.

Building Blocks and How to Use Them

The Directional Valves library offers several prebuilt directional valve models. As indicated in their descriptions, all of them are symmetrical, continuous valves. In other words, the control member in 2-way, 3-way, 4-way, and 6-way valves can assume any position, controlled by the physical signal port S. The valves are symmetrical in that all the orifices the valve is built of are of the same type and size. The only possible difference between orifices is the orifice initial opening.

These configurations cover a substantial portion of real valves, but the directional valves family is so diverse as to make it practically impossible to have a library model for every member. Instead, Simscape Fluids libraries offer a set of building blocks that is comprehensive enough to build a model for any real configuration. This section describes the rules of building a custom model of a directional valve.

All directional valve models are built of variable orifices. In Simscape Fluids libraries, the following variable orifice models are available:

- Annular Orifice
- Orifice with Variable Area Round Holes

- Orifice with Variable Area Slot
- Variable Orifice
- Ball Valve
- Ball Valve with Conical Seat
- Needle Valve
- Poppet Valve

To simplify the way variable orifices are combined in a model, their instantaneous opening is computed in the same way for all types of orifices. The orifice opening is always computed in the direction the spool, or any other control member, opens the orifice. In other words, positive value of the opening corresponds to open orifice, while negative value denotes overlapped, or closed, orifice. The origin always corresponds to zero-lap position, when the edge of the control member coincides with the edge of the orifice. In the illustration below, origins are marked with 0 for the orifice with variable area round holes (schematic on the left) and for the orifice with variable slot (schematic on the right). The x arrow denotes the direction in which orifice opening is measured in both cases.
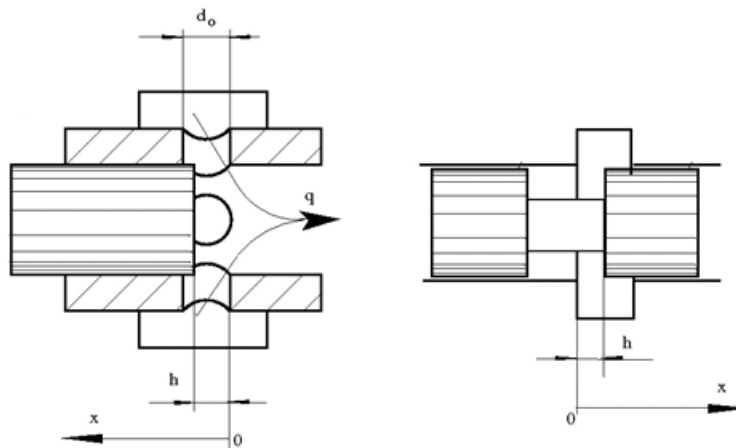
The instantaneous value of the orifice opening is determined as

$$h = x_0 + x_{sp} \cdot or$$

where

h	Instantaneous orifice opening.
x_0	Initial opening. The initial opening value is positive for initially open (underlapped) orifices and negative for overlapped orifices.
x_{sp}	Spool (or other control member) displacement from initial position, which controls the orifice.
or	Orifice orientation indicator. The variable assumes +1 value if a spool displacement in the globally assigned positive direction opens the orifice, and -1 if positive motion decreases the opening.

The number of variable orifices and the way they are connected are determined by the valve design. Usually, the model of a valve mimics the physical layout of a real valve. The illustration below shows an example of a 4-way valve, its symbol, and an equivalent circuit of its Simscape Fluids model.
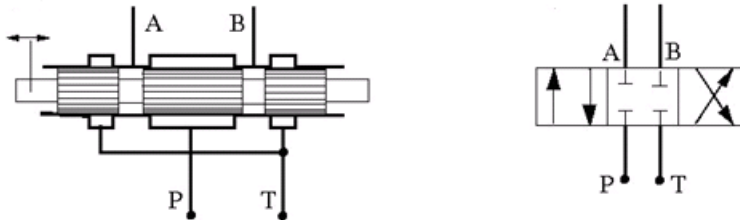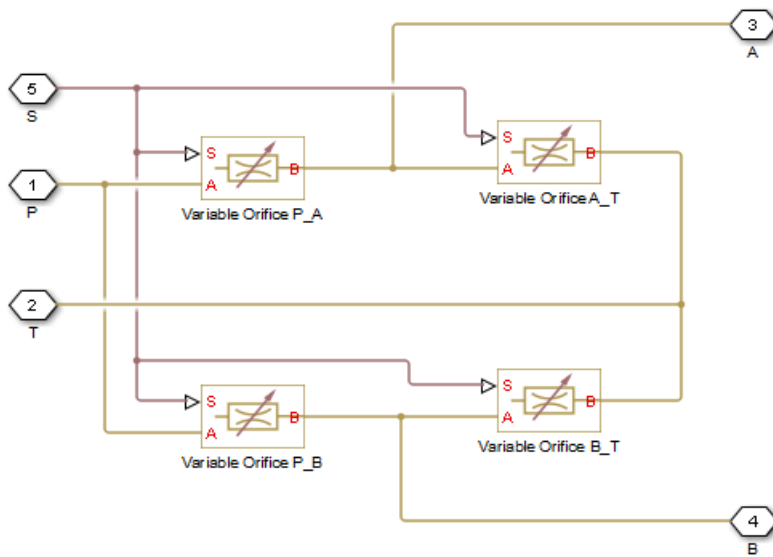
The 4-way valve in its simplest form is built of four variable orifices. In the equivalent circuit, they are named P_A, P_B, A_T, and B_T. The Variable Orifice block, which is the most generic model of a variable orifice in the Simscape Fluids libraries, is used in this particular example. You can use any other variable orifice blocks if the real valve design employs a configuration backed by a stock model, such as an orifice with round holes or rectangular slots, poppet, ball, or needle. In general, all orifices in the model can be simulated with different blocks or with the same block, but with different way of parameterization. For instance, two orifices can be represented by their pressure-flow characteristics, while two others can be simulated with the table-specified area variation option (for details, see the `Variable Orifice` block reference page).

The next example shows another configuration of a 4-way directional valve. This valve unloads the pump in neutral position and requires six variable orifice blocks. The Orifice with Variable Area Round Holes blocks have been used as a variable orifice in this model. Port T1 corresponds to an intermediate point between ports P and T.
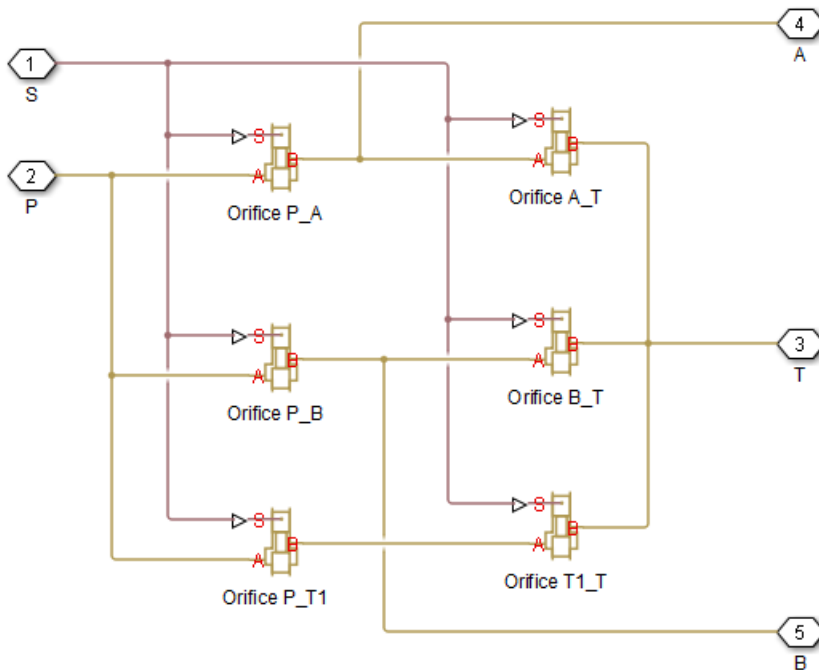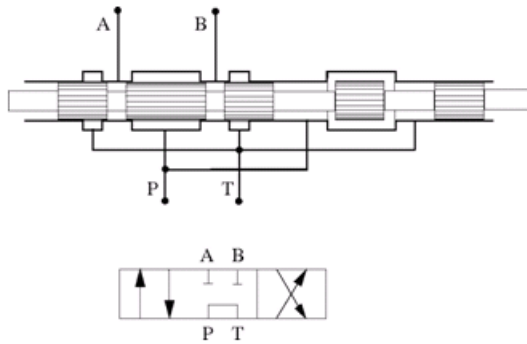
Build a Custom Directional Valve

Finally, let us consider a more complex directional valve example. The figure below shows basic elements of a front loader hydraulic system. Both the lift and the tilt

cylinders are controlled by custom 3-position, 5-way valves, developed for this particular application. The valves are designed in such a way that the pump delivery is diverted to tank (unloaded) if both cylinders are commanded to be in neutral position. The pump is disconnected from the tank if either of the two control valves is shifted from neutral position.
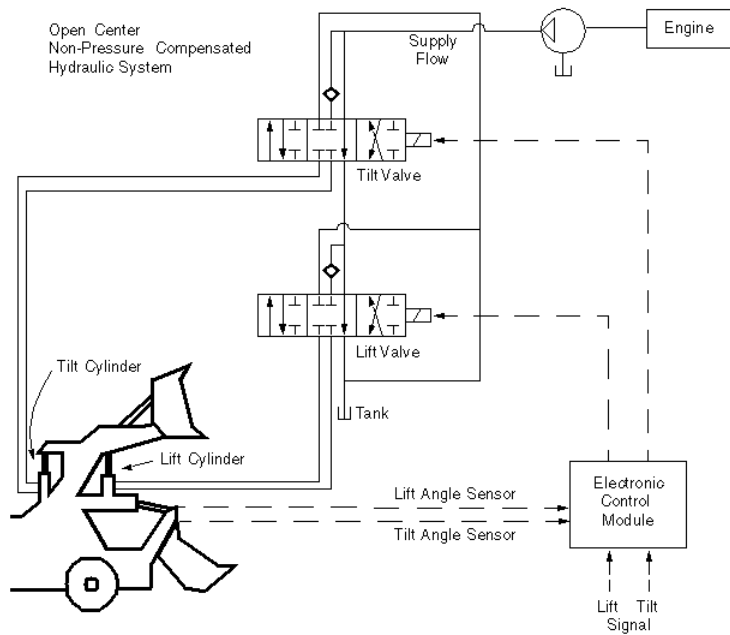
To develop a model, the physical version of the valve must be created first. The following illustration shows one of the possible configurations of the valve.
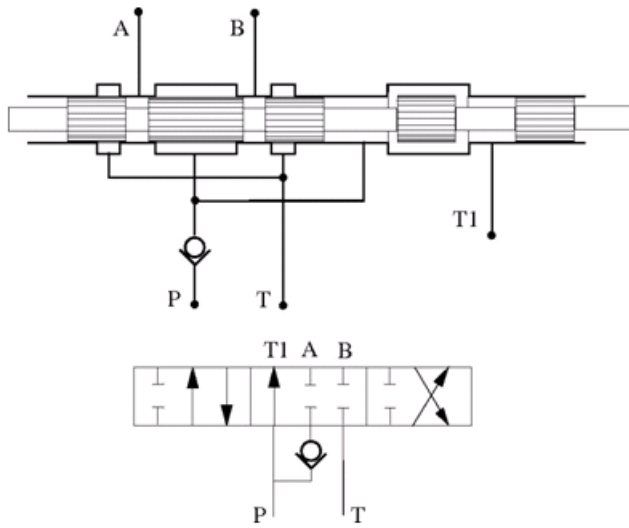
The Simscape Fluids model, shown below, is an exact copy of the physical valve configuration.
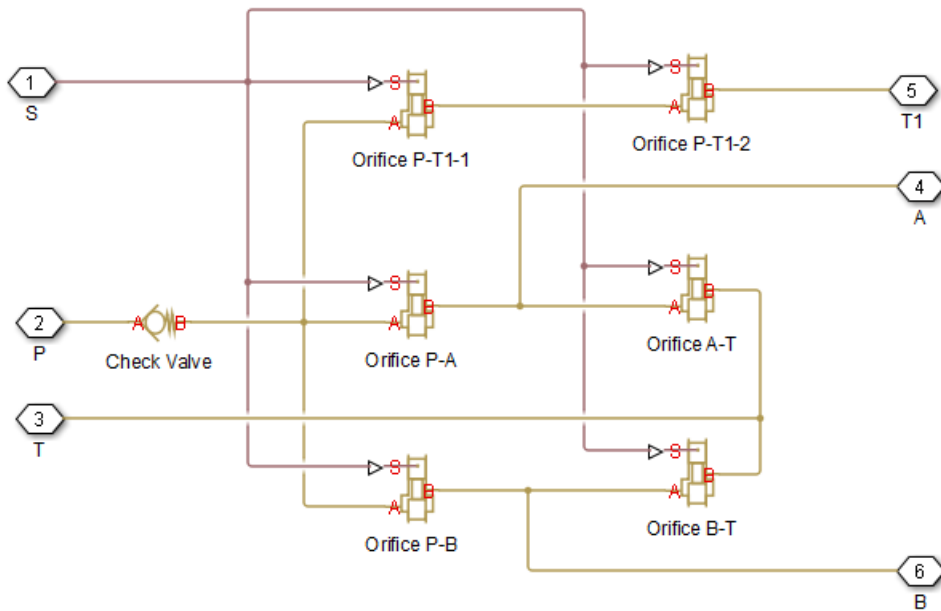
All the orifices in the model are closed (overlapped) in valve neutral position, except orifices P_T1_1 and P_T1_2. These two orifices should be set open to an extent that allows pump delivery to be discharged at low pressure.

Modeling Low-Pressure Fluid Transportation Systems

In this section...
"How Fluid Transportation Systems Differ from Power and Control Systems" on page 2-61
"Available Blocks and How to Use Them " on page 2-63
"Low-Pressure Fluid Transportation System" on page 2-65

How Fluid Transportation Systems Differ from Power and Control Systems

In hydraulics, the steady uniform flow in a component with one entrance and one exit is characterized by the following energy equation

$$-\frac{\dot{W}_s}{\dot{m}g} = \frac{V_2^2 - V_1^2}{2g} + \frac{p_2 - p_1}{\rho g} + z_2 - z_1 + h_L$$

where

$\dot{W}_s$	Work rate performed by fluid
$\dot{m}$	Mass flow rate
V_2	Fluid velocity at the exit
V_1	Fluid velocity at the entrance
p_1, p_2	Static pressure at the entrance and the exit, respectively
g	Gravity acceleration
ρ	Fluid density
z_1, z_2	Elevation above a reference plane (datum) at the entrance and the exit, respectively
h_L	Hydraulic loss

Subscripts 1 and 2 refer to the entrance and exit, respectively. All the terms in Equation 2-1 have dimensions of height and are named kinematic head, piezometric head, geometric head, and loss head, respectively. For a variety of reasons, analysis of hydraulic power and control systems is performed with respect to pressures, rather than to heads, and Equation 2-1 for a typical passive component is presented in the form

$$\frac{\rho}{2}V_1^2 + p_1 + \rho g z_1 = \frac{\rho}{2}V_2^2 + p_2 + \rho g z_2 + p_L$$

where

V_1, p_1, z_1	Velocity, static pressure, and elevation at the entrance, respectively
V_2, p_2, z_2	Velocity, static pressure, and elevation at the exit, respectively
p_L	Pressure loss

Term $\frac{\rho}{2}V^2$ is frequently referred to as kinematic, or dynamic, pressure, and $\rho g z$ as piezometric pressure. Dynamic pressure terms are usually neglected because they are very small, and Equation 2-2 takes the form

$$p_1 + \rho g z_1 = p_2 + \rho g z_2 + p_L$$

The size of a typical power and control system is usually small and rarely exceeds 1.5 – 2 m. To add to this, these systems operate at pressures in the range 50 – 300 bar. Therefore, $\rho g z$ terms are negligibly small compared to static pressures. As a result, Simscape Fluids components (with the exception of the ones designed specifically for low-pressure simulation, described in "Available Blocks and How to Use Them " on page 2-63) have been developed with respect to static pressures, with the following equations

$$p = p_L = f(q)$$
$$q = f(p_1, p_2)$$

where

p	Pressure difference between component ports
q	Flow rate through the component

Fluid transportation systems usually operate at low pressures (about 2-4 bar), and the difference in component elevation with respect to reference plane can be very large. Therefore, geometrical head becomes an essential part of the energy balance and must be accounted for. In other words, the low-pressure fluid transportation systems must

be simulated with respect to piezometric pressures $p_{pz} = p + \rho g z$, rather than static pressures. This requirement is reflected in the component equations

$$p = p_L = f(q, z_1, z_2)$$
$$q = f(p_1, p_2, z_1, z_2)$$

Equations in the form Equation 2-5 must be applied to describe a hydraulic component with significant difference between port elevations. In hydraulic systems, there is only one type of such components: hydraulic pipes. The models of pipes intended to be used in low pressure systems must account for difference in elevation of their ports. The dimensions of the rest of the components are too small to contribute noticeably to energy balance, and their models can be built with the constant elevation assumption, like all the other Simscape Fluids blocks. To sum it up:

- You can build models of low-pressure systems with difference in elevations of their components using regular Simscape Fluids blocks, with the exception of the pipes. Use low-pressure pipes, described in "Available Blocks and How to Use Them" on page 2-63 .

- When modeling low-pressure systems, you must use low-pressure pipe blocks to connect all nodes with difference in elevation, because these are the only blocks that provide information about the vertical locations of the system parts. Nodes connected with any other blocks, such as valves, orifices, actuators, and so on, will be treated as if they have the same elevation.

Available Blocks and How to Use Them

When modeling low-pressure hydraulic systems, use the pipe blocks from the Low-Pressure Blocks library instead of the regular pipe blocks. These blocks account for the port elevation above reference plane and differ in the extent of idealization, just like their high-pressure counterparts:

- Resistive Pipe LP — Models hydraulic pipe with circular and noncircular cross sections and accounts for friction loss only, similar to the Resistive Tube block, available in the Simscape Foundation library.

- Resistive Pipe LP with Variable Elevation — Models hydraulic pipe with circular and noncircular cross sections and accounts for friction losses and variable port elevations. Use this block for low-pressure system simulation in which the pipe ends change their positions with respect to the reference plane.

- Hydraulic Pipe LP — Models hydraulic pipe with circular and noncircular cross sections and accounts for friction loss along the pipe length and for fluid compressibility, similar to the Hydraulic Pipeline block in the Pipelines library.

- Hydraulic Pipe LP with Variable Elevation — Models hydraulic pipe with circular and noncircular cross sections and accounts for friction loss along the pipe length and for fluid compressibility, as well as variable port elevations. Use this block for low-pressure system simulation in which the pipe ends change their positions with respect to the reference plane.

- Segmented Pipe LP — Models circular hydraulic pipe and accounts for friction loss, fluid compressibility, and fluid inertia, similar to the Segmented Pipe block in the Pipelines library.

Use these low-pressure pipe blocks to connect all Hydraulic nodes in your model with difference in elevation, because these are the only blocks that provide information about the vertical location of the ports. Nodes connected with any other blocks, such as valves, orifices, actuators, and so on, will be treated as if they have the same elevation.

The additional models of pressurized tanks available for low-pressure system simulation include:

- Constant Head Tank — Represents a pressurized hydraulic reservoir, in which fluid is stored under a specified pressure. The size of the tank is assumed to be large enough to neglect the pressurization and fluid level change due to fluid volume. The block accounts for the fluid level elevation with respect to the tank bottom, as well as for pressure loss in the connecting pipe that can be caused by a filter, fittings, or some other local resistance. The loss is specified with the pressure loss coefficient. The block computes the volume of fluid in the tank and exports it outside through the physical signal port V.

- Variable Head Tank — Represents a pressurized hydraulic reservoir, in which fluid is stored under a specified pressure. The pressurization remains constant regardless of volume change. The block accounts for the fluid level change caused by the volume variation, as well as for pressure loss in the connecting pipe that can be caused by a filter, fittings, or some other local resistance. The loss is specified with the pressure loss coefficient. The block computes the volume of fluid in the tank and exports it outside through the physical signal port V.

- Variable Head Two-Arm Tank — Represents a two-arm pressurized tank, in which fluid is stored under a specified pressure. The pressurization remains constant regardless of volume change. The block accounts for the fluid level change caused by the volume variation, as well as for pressure loss in the connecting pipes that can be caused by a filter, fittings, or some other local resistance. The loss is specified with the

pressure loss coefficient at each outlet. The block computes the volume of fluid in the tank and exports it outside through the physical signal port V.

- Variable Head Three-Arm Tank — Represents a three-arm pressurized tank, in which fluid is stored under a specified pressure. The pressurization remains constant regardless of volume change. The block accounts for the fluid level change caused by the volume variation, as well as for pressure loss in the connecting pipes that can be caused by a filter, fittings, or some other local resistance. The loss is specified with the pressure loss coefficient at each outlet. The block computes the volume of fluid in the tank and exports it outside through the physical signal port V.

Low-Pressure Fluid Transportation System

The following illustration shows a simple system consisting of three tanks whose bottom surfaces are located at heights H1, H2, and H3, respectively, from the reference plane. The tanks are connected by pipes to a hydraulic manifold, which may contain any hydraulic elements, such as valves, orifices, pumps, accumulators, other pipes, and so on, but these elements have one feature in common – their elevations are all the same and equal to H4.
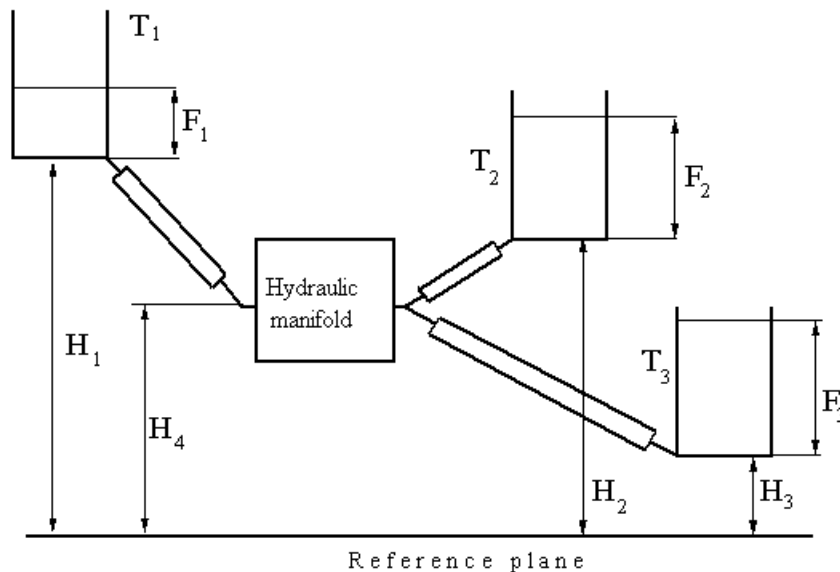
The models of tanks account for the fluid level heights F1, F2, and F3, respectively, and represent pressure at their bottoms as

$$p_i = \rho g F_i \quad \text{for } i = 1, 2, 3$$

The components inside the manifold can be simulated with regular Simscape Fluids blocks, like you would use for hydraulic power and control systems simulation. The pipes must be simulated with one of the low-pressure pipe models: Resistive Pipe LP, Hydraulic Pipe LP, or Segmented Pipe LP, depending on the required extent of idealization. Use the Constant Head Tank or Variable Head Tank blocks to simulate the tanks. For details of implementation, see the Water Supply System and the Three Constant Head Tanks examples.