Tutorial 2

Hydraulic Cylinder Drive

Objective
In this tutorial you will create a model for a simple hydraulic cylinder drive, which is controlled by a proportional directional control valve. In a first step you will create an Open-Loop System, which means that there is no feedback of the actual cylinder position to the control valve. In a second step, the position of the cylinder is measured and compared to the position command signal, in order to create a Closed-Loop System.

The purpose of the circuit is to lift (or lower, respectively) the load mass in vertical direction, according to the command signal of the proportional directional control valve. For the pressure supply, we simply use a constant displacement pump, driven at a constant speed and connected to a pressure relief valve. The advantage of this system is its simple structure. Of course, it shows a very poor efficiency, since a great amount of energy is consumed by the flow through the relief valve.

It is assumed that you are familiar with the basic functionality of SimulationX. Therefore, please refer to “Tutorial 1: Getting Started” for a general introduction on how to select Elements from the libraries, how to connect them and enter parameters, how to run a simulation and how to open result windows.

Part 1: Open-Loop System
Create the SimulationX model of the hydraulic cylinder drive according to Figure 1. Use the elements listed in Table 1. You can change the label text and label position of each object by double clicking on an element and selecting “General/Name…”. You have to write the name without a blank (e.g. “CylinderDrive”).

When connecting the elements with each other, you should remember, that in SimulationX you can only connect element ports of the same type.

The mass of the cylinder piston is not included in the cylinder object. So you have to model the piston mass by connecting an element “Mass” (Library Mechanics→Linear Mechanics→Mass) to the cylinder. It represents the piston mass as well as the load mass at the same time.

In order to cut an already existing connection, click on it and press “Del”.

The path of a connection will be determined automatically, but changing the path is possible at any time. To do this, move the mouse over a connection, while pressing the Alt-key. The mouse pointer shows you in which direction you can move the selected connection line.
Figure 1: Model structure of the Open-Loop Cylinder Drive

Table 1: Elements required for the circuit in Figure 1

<table>
<thead>
<tr>
<th>Number of elements</th>
<th>Library name</th>
<th>Element name</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Hydraulics/Actuators</td>
<td>diffCylinder1</td>
<td>![DiffCylinder1]</td>
</tr>
<tr>
<td>1</td>
<td>Hydraulics/Actuators</td>
<td>pumpMotor1</td>
<td>![PumpMotor1]</td>
</tr>
<tr>
<td>1</td>
<td>Hydraulics/Basic Elements</td>
<td>volume1</td>
<td>![Volume1]</td>
</tr>
<tr>
<td>1</td>
<td>Hydraulics/Valves/Pressure Valves</td>
<td>pressureReliefValve1</td>
<td>![PressureReliefValve1]</td>
</tr>
<tr>
<td>1</td>
<td>Hydraulics/Basic Elements</td>
<td>tank1</td>
<td>![Tank1]</td>
</tr>
</tbody>
</table>
Once you have succeeded in creating the model structure according to Figure 1, you have to enter the parameters for the elements and activate the protocol attribute for the variables which you want to plot after the simulation. Since almost all elements have default parameters, you only have to enter the parameters which are different from the default values. Table 2 gives an overview about the parameter settings which you have to enter. Some of the parameter dialogs have more than one page.

Table 2: Parameter settings for the elements listed in Table 1

<table>
<thead>
<tr>
<th>Element</th>
<th>Parameter Input</th>
</tr>
</thead>
</table>
| Cylinder (diffCylinder1) | Dialog Page “Geometry” :
  - Set the Maximum Cylinder Stroke to 400 mm
  - Set the “Dead Volume” at Port A and B to 1 dm³ (select appropriate unit first and enter the value secondly)
  - Set the “Transformation of Piston Housing” to -200 mm
    - Transfer of Coordinates: dXh : -200
      - The coordinate transformation dXh of -200 mm means, that the displacement of the “Mass” is zero while the piston stroke is 200 mm. It sets the center of the mass to the half of the piston stroke.
  - Dialog Page “Friction” :
    - Friction Description: No Friction Losses
    - Dialog Page “Results 2” :
      - Activate the result attributes

<table>
<thead>
<tr>
<th>Element</th>
<th>Parameter Input</th>
</tr>
</thead>
</table>
Mass (mass1)

- Set the mass to 100 kg

GravityForce (source1)

- Set the force to (mass1.m*9.81) N

Note, that in SimulationX you can enter constant values as well as arithmetical and/or Boolean expressions, variables and functions for any parameter. In the case above, the expression considers the gravity force of mass1.m (100) kg.

When using the "External Force" element, you have to decide whether to connect the left or the right side to the mechanical element (i.e. mass). In our case, we have connected the left side of the force to the mass (see Figure 1). This means, that the force will act against the positive direction of motion of the mass. The small red arrows indicate the positive coordinate direction of the element (mass), the positive coordinate direction of the force is indicated by the big white arrows.

- Activate the protocol attributes for the force

Pump (pumpMotor1)

Dialog page “Geometry”:
- Set the displacement volume to 50 cm³

Dialog page “Friction”:

Dialog page “Leakage”:

Dialog page “Results2”:
- Activate the protocol attribute for the flow at port A

Motor (preset1)

Dialog page “Parameters”:
- Select the kind “Rotational Speed”
- Select at first the unit “rpm” and then the value for the rotational speed
- Activate the protocol attributes for “Torque” and “Power”
ReliefValve (pressureReliefValve1)

Activate the protocol attribute for the result variables “Pressure Drop”, “Volume Flow” and “Power Dissipation”

ProportionalControlValve (propDirValve1)

Dialog page “Stroking”:
- In the selection box “Stroke Signal” select “Normalized Signal”

Note, that “Normalized Signal” means the valid range of the input signal must be from –1 to +1. At an input signal of zero the valve will be in its center position.

Dialog page “Dynamics”:
- Turn off the check box for the valve dynamics

Dialog page “Q-y-Function”:
- In the selection box “Type of Edges” select “Identical Edges”
- Set the flow per change of Stroke to (60 l/min)/-

Change the unit to (l/min)/- firstly.

Note: A value of 60 (l/min)/- means, that we will have a flow of 60l/min at a pressure drop of 35 bar (at one single edge) for full valve opening stroke.

Dialog page “Results”:
- Activate the protocol attribute for the “Relative Valve Stroke”
- Activate the protocol attributes for the flow at port A & B

ControlSignal (curve1)

Dialog “Parameters”:
- Select “Simulation Time t [s]”
• Click the “Edit” button to open the dialog for the “Input of Values”

• Select the quantity and the labels for the axis by using the “Properties” button in the toolbar

In the “Properties”- window you have to enter the comments and the quantities for the axes. Enter for the X-axis the simulation time and for the Y-axis as comment “Signal Output” and quantity “Base Quantities/ Relative Magnitude”.

• Enter the following data and click OK when finished:

You can also load existing data from ASCII files or double-click directly on the graphic on the right. Alternatively, you can use other elements from the library “Signal Sources”.

• Activate the protocol attribute for the result variable “Signal Output”.

VP (volume1)

Dialog page “Parameters”:

• Set the volume to 1 dm³

Note, that volumes are not required in SimulationX. However a pump volume of zero would result in an infinitely fast pressure change, which is not realistic.

Dialog page “Results”:

• Activate the protocol attribute for the pressure
At the beginning the load of the piston is hold at a defined point. That means this connection must have an initial pressure. The pressure force acts against the gravity force.

- Set the initial pressure $p_0$ to the following term:

$$ \begin{align*}
\text{Initial Pressure (relative to pAtm)} = p_0: & \quad \text{source1.F/diffCylinder1.area} \\
\end{align*} $$

Let the settings of simulation as default. Once you have entered the parameters of the model, you can run the simulation and observe the results. Figure 2 shows some of the simulation results. You can also change the hydraulic fluid by double-clicking on the connection and selecting a liquid. The default is "HLP 46", a mineral oil with a viscosity behavior according to ISO VG 46.

The control signal shows the specified characteristics. If the stroke signal for the valve is negative, the pump is connected to Port A of the cylinder, i.e. the load mass is lifted up.

The flow at Port A and B of the valve is somehow proportional to the stroke signal. Because we have a differential cylinder, the flow shows an unsymmetrical behavior. The maximum positive flow can not exceed 50 l/min, which is the flow of the pump.

The pressure in volume “VP” can not exceed 100 bar, which is the set pressure for the relief valve. At full negative opening of the control valve, the pump pressure “VP” drops to 32 bar, since the pump can not deliver enough flow.
The velocity of the piston is proportional to the flow in the valve. As the pump pressure drops from 100 bar to 32 bar, the velocity also decreases slightly. When the control valve is closed, the piston oscillates because of the oil compressibility.

You can play with the simulation model, in order to improve the performance of the open-loop cylinder drive. (Don’t forget to reset the model before you make modifications!) Some interesting changes in the behavior of the system should become obvious, if you

- change the set pressure in the relief valve $p_{\text{Set}}$ from 100 bar to 200 bar, and
- use a smaller proportional control valve (i.e. change the “Flow per Change of Stroke” from 60 (l/min)/- to 20 (l/min)/-)

Especially with the last modification, the pump pressure will remain constant since the flow demand of the cylinder is now decreased.
Part 2: Adding an Accumulator

You will now add an accumulator to the system in order to compensate for short flow demands from the cylinder, which exceed the flow of the pump. Before you will do this, please reset the parameter "Flow per Change of Stroke" of the Proportional Control Valve from 20 (l/min)/- to 60 (l/min)/- and the set pressure in the relief valve to 100 bar.

To add the accumulator to the model, carry out the following steps:

- Reset the simulation model.
- Delete the volume VP.
- Open the library “Hydraulics” (Accessories and Sensors) and connect the “Hydropneumatic Accumulator” according to Figure 4:

Table 3: Accumulator parameters

<table>
<thead>
<tr>
<th>Accu (accumulator1)</th>
<th>Dialog page “Construction 1”:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Set construction type to “Bladder Accumulator (vertical)”</td>
</tr>
<tr>
<td></td>
<td>Construction Type geomType: Bladder Accumulator (vertical)</td>
</tr>
<tr>
<td></td>
<td>Set the accumulator gas volume to 5 dm³</td>
</tr>
</tbody>
</table>

Figure 4: Modified model structure with accumulator
- Set the “Pre-Fill Pressure” to 80 bar

Pre-Fill Pressure \( p_{\text{GasF}}: 80 \) bar

- Set the dead volume at oil-side to 100 cm³

Dead Volume at Oil-Side \( V_{\text{Oil}}: 100 \) cm³

Dialog page “Operating Conditions”:
- Set the “Initial Oil Pressure” to 100 bar

Initial Oil Pressure \( p_{\text{OilO}}: 100 \) bar

Dialog page “Results 1”:
- Activate the protocol attribute for “Oil Pressure”

\[ \text{Oil Pressure (relative to } \ldots \text{pOil)} : \quad \text{bar} \]

- Activate the protocol attribute for “Volume Flow of Oil”

\[ \text{Volume Flow of Oil int... QOil:} \quad \text{dm}^3/\text{min} \]

- Activate the protocol attribute for “Gas Pressure”

\[ \text{Gas Pressure (relative to } \ldots \text{pGas):} \quad \text{bar} \]

- Activate the protocol attribute for “Gas Volume”

\[ \text{Gas Volume} \quad V_{\text{Gas}}: \quad \text{dm}^3 \]

If you run the simulation, you will find that the cylinder reaches its end stop after 0.3 s. At this moment, the velocity of the piston starts to oscillate with a very high frequency, because of the high end-stop stiffness of the cylinder.

![Figure 5: Improved velocity with the accumulator](image)

It would now be possible to adapt the stiffness and the damping of the cylinder end stop to the force of the cylinder and to the piston and load mass. However, you will rather make the system a closed-loop cylinder drive.
Part 3: Closed-Loop System

For many industrial applications, an open-loop cylinder drive is not sufficient. Closed-loop systems are used in such cases. You can easily enhance the given example of the cylinder drive to make it a closed-loop drive. Add the new objects as shown in Figure 6 and listed in Table 4.

![Closed-loop cylinder drive](image)

**Table 4:** New objects needed for the closed-loop system

<table>
<thead>
<tr>
<th>Object</th>
<th>Library</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor (sensor1)</td>
<td>Mechanics</td>
<td>None</td>
</tr>
<tr>
<td>PistonError (sum1)</td>
<td>Signal Blocks</td>
<td>Set the sign switches as follow:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>x1 x2 x3 Var:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>X1Var: - X2Var: + X3Var: -</td>
</tr>
</tbody>
</table>

Note, that for controlling the process we have to compare the real displacement with the given displacement.
As the next step, you have to adapt the data table for the command signal from a relative quantity (-) to a displacement signal (mm). To do this, open the parameter dialog for the model object "CommandSignal" and proceed as follows:

- **Open the curve editor window by clicking on the "Edit"-Button in the dialog "Parameters"**
  
  ![Curve editor window](image1.png)

- **Edit the unit by clicking on "Properties"-Button**

Define the quantity of the signal output as displacement:

![Properties window](image2.png)

Adapt the values of the table to a maximum of 100 mm:

![Data table](image3.png)

Close the curve editor window with OK.

Now you have finished the parameter entry and we can run the simulation.
You have changed the signal from a command signal to a position signal. It means another behavior as before, because you now control the position and not the signal for the ProportionalControlValve.

It shows the result of the piston displacement compared to the value of the command signal.
It can be observed, that the sign of the gain is OK but the value of 1 is too small.

If you run the simulation again, you can observe that the performance has improved considerably.

However if you open the result window for the mass velocity, you will see that the system tends to be unstable.

If such an effect occurs in reality, the gain must be decreased. However, in your model you have neglected the dynamic behavior of the control valve. Therefore, you have to include the dynamics of the valve by editing the following parameters of the "ProportionalControlValve":

![Figure 7: Simulation Result for the closed-loop cylinder drive with a gain of 1](image)

So let us increase the gain of the signal block "Gain" from 1 to 30:

![Figure 8: Simulation Result with a gain of 30](image)
Table 5: New parameters of “ProportionalControlValve”

<table>
<thead>
<tr>
<th>ProportionalControlValve</th>
<th>Activate the valve dynamics:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>![Checkbox] Consider Valve Dynamics</td>
</tr>
</tbody>
</table>

- Set the undamped “Natural Frequency” to 18 Hz and the “Damping Ratio” to 0.8:

<table>
<thead>
<tr>
<th>Natural Frequency (Undamp.)</th>
<th>18 Hz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Damping Ratio</td>
<td>0.8</td>
</tr>
</tbody>
</table>

If you run the simulation again with the valve dynamics included, you can observe that the system shows now a stable behavior – see Figure 9.

![Figure 9: Stable behavior of the mass velocity at a gain of 30 with the valve dynamics included in the model](image)

You can play with the simulation model in order to include additional physical effects in the simulation (e.g. friction or leakage in the cylinder) or to optimize the control strategy (e.g. by replacing the simple P-Gain block by a PID block). Alternatively, an Optimization Tool could be used to find optimum values for the controller.

Finally, let us resume a few points concerning the benefits of this tutorial:

- You have learned how to construct models of hydraulic drive systems in SimulationX. Most of the elements have default parameters and scalable options (e.g. cylinder friction and leakage).
- Signal blocks in SimulationX (e.g. signal sources) can be adapted to your purpose in terms of physical quantities, units and names of parameters and result quantities. Complex data tables can be entered or even imported from external files.
- SimulationX is a tool for intuitive system simulation, since models with mixed physical domains (e.g. mechanics, hydraulics and controls) can be created very quickly.
- The object-oriented approach of SimulationX allows you to modify a given model structure very easily and adapt it to new tasks and specifications.
- The SimulationX model clearly resembles the circuit structure of the hydraulic system. This means, that you can concentrate on your engineering task instead of thinking about the mathematical background of your system. Moreover, other engineers can easily re-use and understand your own models.