Ackumulator

```
<< C:\\Hopsan\Compgen\CompgenNG.mx
```

In[73]:=
```
path = ToFileName[{"C:", "HopsanTrunk",
     "ComponentLibraries", "defaultLibrary", "Hydraulic", "Volumes&Lines"}];

domain = "Hydraulic";
displayName = "Ackumulator";
brief = "This is piston with an inertia load";
componentType = "ComponentQ";
author = "Petter Krus <petter.krus@liu.se>";
affiliation =
  "Division of Fluid and Mechatronic Systems, Linköping University";
SetFilenames[path, domain, displayName];
ResetComponentVariables[];
Date[]
```

```
{2014, 12, 11, 11, 27, 21.3455121}
```

```
file
```

```
C:\HopsanTrunk\ComponentLibraries\defaultLibrary\Hydraulic\Volumes&Lines\
  HydraulicAckumulator.hpp
```

Component description

A general ackumulator. This model does not have any thermodynamic losses. The process is considered to be adiabatic.

Variables and parameters

```
inputParameters = {
    {V0, 0.001, double, "m^3", "Ackumulator Volume"},
    {Kca, 1. × 10^-8, double, "m^3/(s Pa)", "Ack. inlet coeff."},
    {kappa, 1.2, double, "", "polytropic exp. of gas"},
    {p0, 1. * 10^7, double, "N/m^2", "Preload pressure"}
  };

outputVariables = {
    {Va, 1. * 10^-3, double, "m^3", "Momentary gas volume"},
    {pa, 1. * 10^7, double, "Pa", "Ackumulator oil pressure"},
    {xmp, 0., double, "", "State of charge (Set startvalue here!)"},
    {vmp, 0., double, "", "State of charge speed"}
  };
```

```
nodeConnections = {
      HydraulicQnode[1, 1. * 10^5, "hydraulic node 1"]};
```

The system of equations

Using the equations for piston ackumulator by setting stroke to one.

```
SL = 1;
```

```
Ap = V0 / SL;
```

The restriction in the inlet is recalculated as a viscous friction on the “piston”.

$$\text{Bp} = \frac{\text{Ap}^2}{\text{Kca}};$$

The generated force on the "piston"

```
fg = Ap p1 - Ap pa;
```

```
systemEquationsDA := {
  Bp vmp == fg,
  Bp der[xmp] == fg ,
  q1 == - Ap vmp,
  pa (limit[SL - xmp, 0.1 SL, SL] Ap)^kappa == p0 (SL Ap)^kappa
    }
```

```
expressions =
  {Va == (SL - xmp) Ap};
```

Limitatons

```
variable2Limits = {{xmp, vmp, 0., SL}};
```

```
variableLowLimits = {{p1, 0.}};
```

The boundarys

```
systemBoundaryEquations = {
   p1 == c1 + Zc1e q1
  };
```

The vector of independent variables of the system are

```
systemVariables = {vmp, xmp, q1, pa, p1};
```

```
Compgen[file]
```

```
Bp =.; Ap =.; SL =.;
```