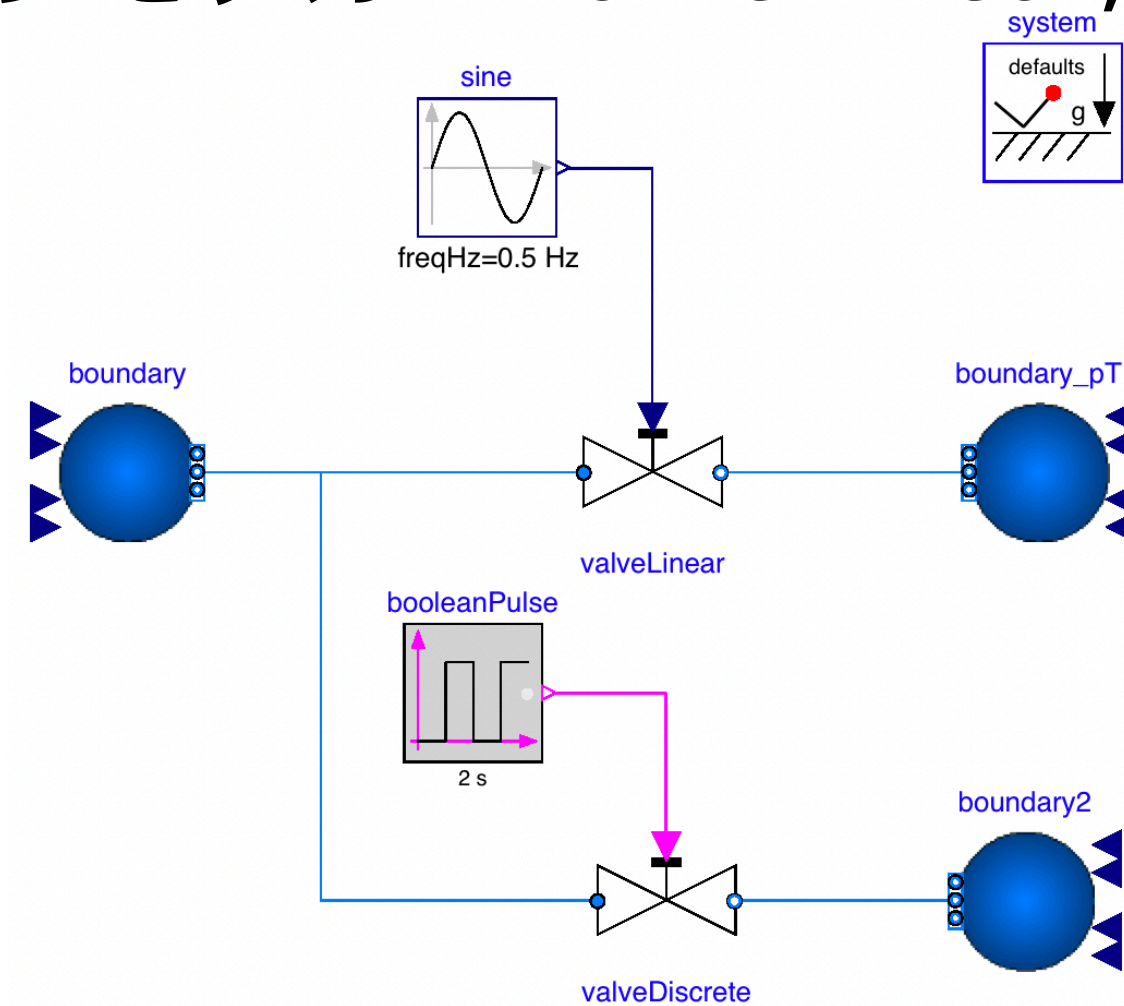


Modelica.Fluid.Valves について (1)

簡易バルブモデル ValveLinear, ValveDiscrete



第14回 Modelica ライブラリ勉強会

2019/09/28 finback

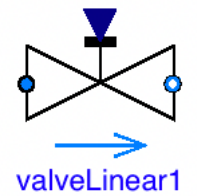
簡易バルブモデル

Modelica Standard Library (MSL) の Modelica.Fluid.Valves に含まれる **ValveLinear** と **ValveDiscrete** は、バルブの流量が圧力差と開度に比例する簡易バルブモデルです。全開時の**基準圧力**(dp_nominal)と**基準流量**(m_flow_nominal)を設定することによって比較的手軽に使用できます。

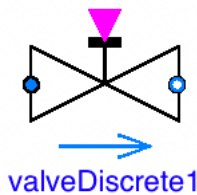
Contents

- Modelica.Fluid.Valves のバルブモデル
- 簡易バルブモデルの継承関係
- PartialTwoPortTransport
- ValveLinear
- ValveDiscrete
- 例題 ValveExample1 SimpleValve1
- まとめ

Modelica.Fluid.Valves のバルブモデル

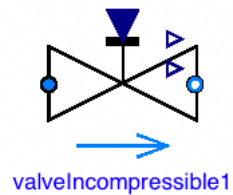


ValveLinear
流量と開度・圧力差が比例する

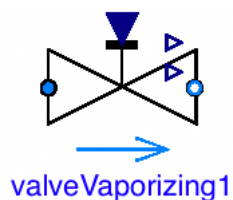


ValveDiscrete
流量と圧力差が比例する
全開と全閉のみ

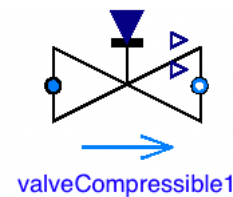
簡易バルブモデル



ValveIncompressible
液体用バルブ



ValveVaporizing
内部で液体が蒸発するバルブ



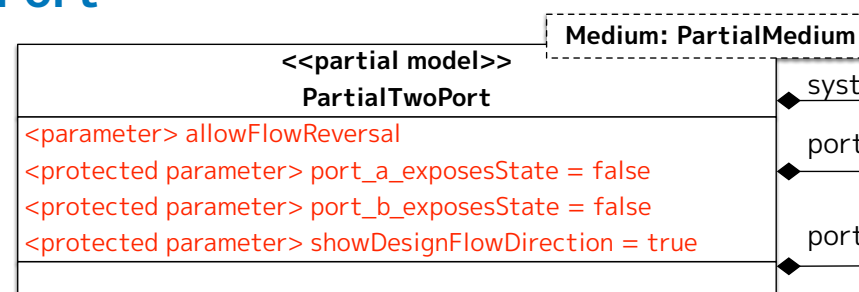
ValveCompressible
気体用バルブ

バルブ選定用の
性能データを用いたモデル

モデルの継承関係

Modelica.Fluid.Interfaces.PartialTwoPort

出入口(2個のFluidPort)
をもつコンポーネントの
ベースモデル



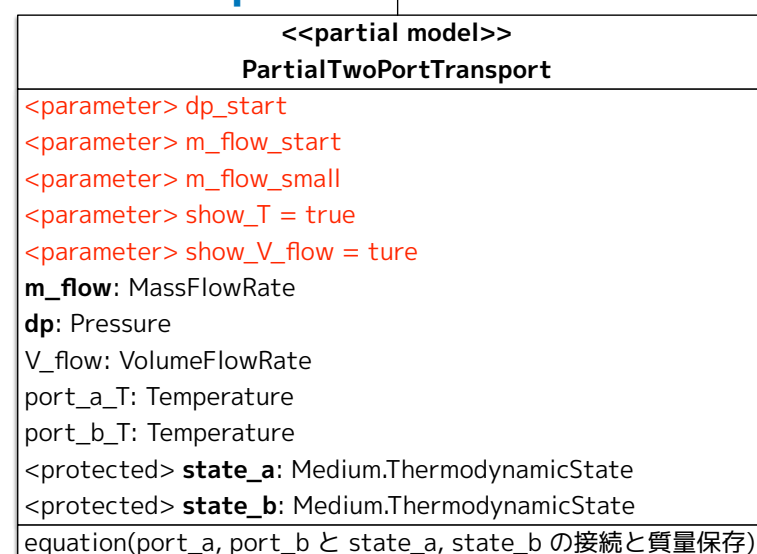
流体物性

FluidPort

質量流量 m_{flow}
圧力 p
エンタルピー $h_{outflow}$
質量分率 $Xi_{outflow}$
微小物質 $C_{outflow}$

Modelica.Fluid.Interfaces.PartialTwoPortTransport

内部に流体を蓄えない
コンポーネントの
ベースモデル

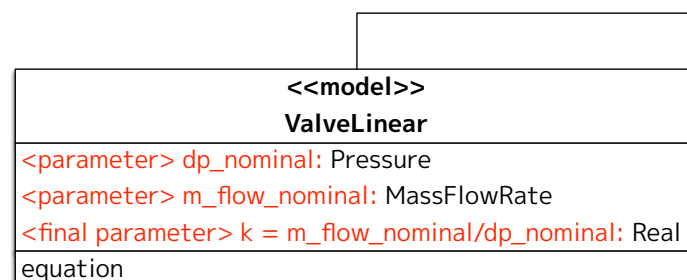


基礎方程式

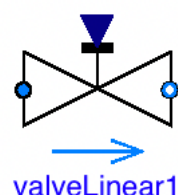
- ①物性モデル式
- ②質量保存則
- ③圧力流量関係式
- ④エネルギー保存式

Modelica.Fluid.Valves.ValveLinear

- ③ 圧力流量関係式
- ④ エネルギー保存式

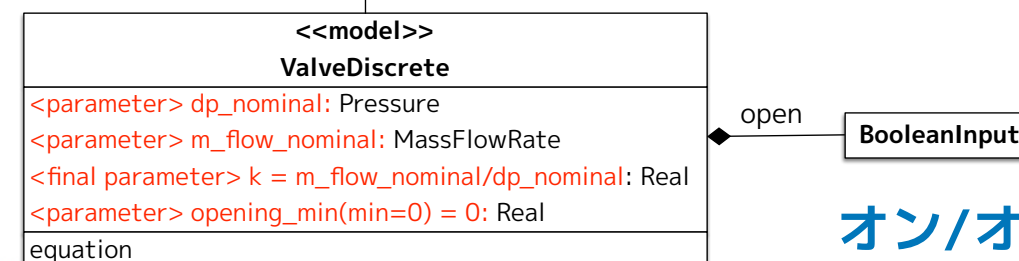


バルブ開度

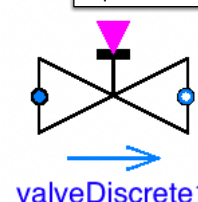


流量と開度・圧力差が比例する

Modelica.Fluid.Valves.ValveDiscrete



オン/オフ



- ・ 流量と圧力差が比例する
- ・ 全開と全閉のみ

PartialTwoPortTransport

PartialTwoPortTransport の方程式

```
equation
// medium states
state_a = Medium.setState_phX(port_a.p, inStream(port_a.h_outflow), inStream(port_a.Xi_outflow));
state_b = Medium.setState_phX(port_b.p, inStream(port_b.h_outflow), inStream(port_b.Xi_outflow));

// Pressure drop in design flow direction
dp = port_a.p - port_b.p;

// Design direction of mass flow rate
m_flow = port_a.m_flow;
assert(m_flow > -m_flow_small or allowFlowReversal, "Reversing flow occurs even though allowFlowReversal is false");

// Mass balance (no storage)
port_a.m_flow + port_b.m_flow = 0;

// Transport of substances
port_a.Xi_outflow = inStream(port_b.Xi_outflow);
port_b.Xi_outflow = inStream(port_a.Xi_outflow);

port_a.C_outflow = inStream(port_b.C_outflow);
port_b.C_outflow = inStream(port_a.C_outflow);

annotation ( ... );
end PartialTwoPortTransport;
```

① 流入時の物性モデル
(状態方程式を含む)

$$\Delta p = port_a.p - port_b.p \quad \text{出入り口の圧力差}$$
$$\dot{m} = port_a.m_flow \quad \text{質量流量}$$

② 質量保存則 (流入量=流出量)

体積流量とポートの温度

```

Modelica.SIunits.VolumeFlowRate V_flow=
  m_flow/Modelica.Fluid.Utilities.regStep(m_flow,
    Medium.density(state_a),
    Medium.density(state_b),
    m_flow_small) if show_V_flow
  "Volume flow rate at inflowing port (positive when flow from port_a to port_b)";

Medium.Temperature port_a_T=
  Modelica.Fluid.Utilities.regStep(port_a.m_flow,
    Medium.temperature(state_a),
    Medium.temperature(Medium.setState_phX(port_a.p, port_a.h_outflow, port_a.Xi_outflow)),
    m_flow_small) if show_T
  "Temperature close to port_a, if show_T = true";

Medium.Temperature port_b_T=
  Modelica.Fluid.Utilities.regStep(port_b.m_flow,
    Medium.temperature(state_b),
    Medium.temperature(Medium.setState_phX(port_b.p, port_b.h_outflow, port_b.Xi_outflow)),
    m_flow_small) if show_T
  "Temperature close to port_b, if show_T = true";

```

$$V_{flow} = \frac{\dot{m}}{\rho}$$

体積流量

$$\rho = \text{regStep}(\dot{m}, \rho_a, \rho_b, \dot{m}_{small})$$

密度

port_a の温度

$$\text{port_a_T} = \text{regStep}(\dot{m}, \text{流入時の温度}, \text{流出時の温度}, \dot{m}_{small})$$

流出時の温度

port_b の温度

$$\text{port_b_} = \text{regStep}(\dot{m}, \text{流入時の温度}, \text{流出時の温度}, \dot{m}_{small})$$

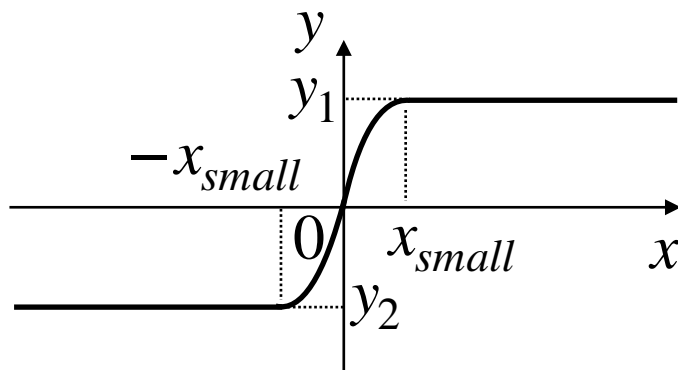
流出時の温度

$$\text{regStep}(x, y1, y2, x_{small})$$

```

algorithm
  y := smooth(1, if x > x_small then y1 else
    if x < -x_small then y2 else
      if x_small > 0 then (x/x_small)*((x/x_small)^2 - 3)*(y2-y1)/4 + (y1+y2)/2 else (y1+y2)/2);
  annotation(Documentation(revisions= "<html> ...));
end regStep;

```



- 温度や密度は上流側の値を使う。
- 流量が反転するときに不連続に変化しないようにする。

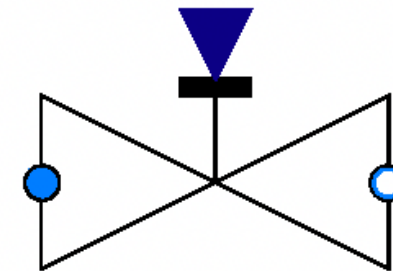
ValveLinear 流量が圧力差と開度に比例するバルブ

③圧力流量関係式

$$\dot{m} = \text{opening} \cdot k \cdot \Delta p$$

$$k = \frac{\dot{m}_{nominal}}{\Delta p_{nominal}}$$

$$0 \leq \text{opening} \leq 1$$



$\dot{m}_{nominal}$ = m_flow_nominal [kg/s] 基準流量（全開時の流量）

$\Delta p_{nominal}$ = dp_nominal [bar] 基準圧力差

比例係数

```
final parameter Types.HydraulicConductance k = m_flow_nominal/dp_nominal
```

valveLinear の方程式

```
equation
```

```
  m_flow = opening*k*dp;
```

```
  // Isenthalpic state transformation (no storage and no loss of energy)
```

```
  port_a.h_outflow = inStream(port_b.h_outflow);
```

```
  port_b.h_outflow = inStream(port_a.h_outflow);
```

④エネルギー保存則（エンタルピー変化無し）

```
annotation ( ... );
```

```
end ValveLinear;
```

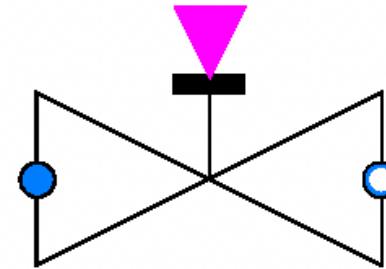

ValveDiscrete オン/オフ制御バルブ

③圧力流量関係式

$$\dot{m} = \begin{cases} k\Delta p, & \text{open} = true \\ opening_min \cdot k\Delta p, & \text{open} = false \end{cases}$$

$$k = \frac{\dot{m}_{nominal}}{\Delta p_{nominal}}$$

$$\text{open} = \begin{cases} true \\ false \end{cases}$$



$\dot{m}_{nominal}$ = m_flow_nominal [kg/s] 基準流量（全開時の流量）

$\Delta p_{nominal}$ = dp_nominal [bar] 基準圧力差

比例係数

$opening_min$ = 最小開度 [-]

```
final parameter Types.HydraulicConductance k = m_flow_nominal/dp_nominal
```

valveDiscrete の方程式

equation

```
m_flow = if open then 1*k*dp else opening_min*k*dp;
```

```
// Isenthalpic state transformation (no storage and no loss of energy)
```

```
port_a.h_outflow = inStream(port_b.h_outflow);
```

```
port_b.h_outflow = inStream(port_a.h_outflow);
```

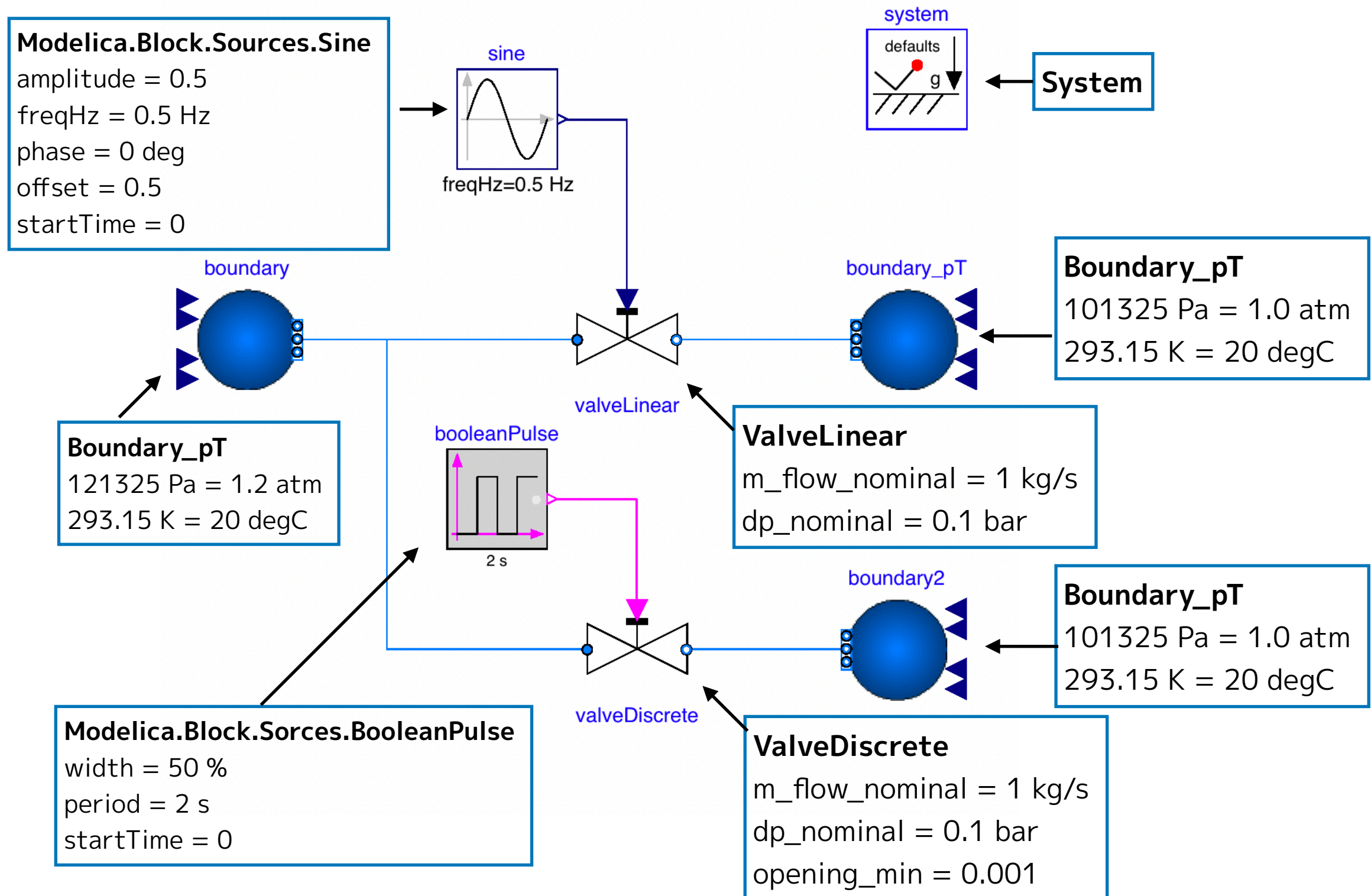
④ エネルギー保存則（エンタルピー変化無し）

```
annotation ( ... );
```

```
end ValveDiscrete;
```


例題 ValveExample1 SimpleValve1

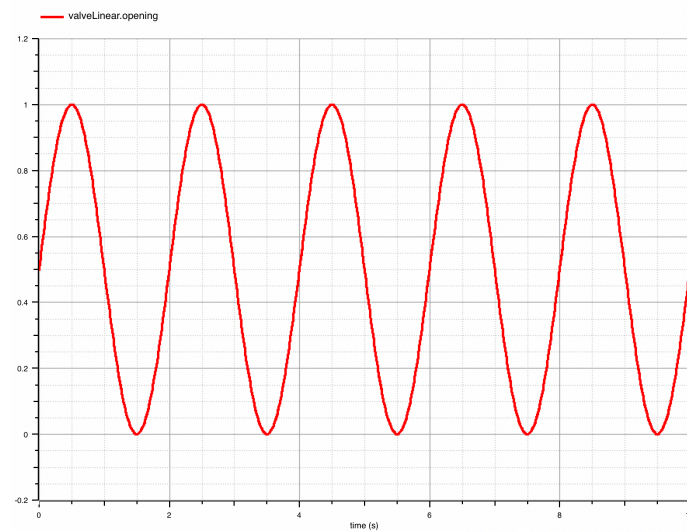
Medium = Modelica.Media.Water.StandardWater



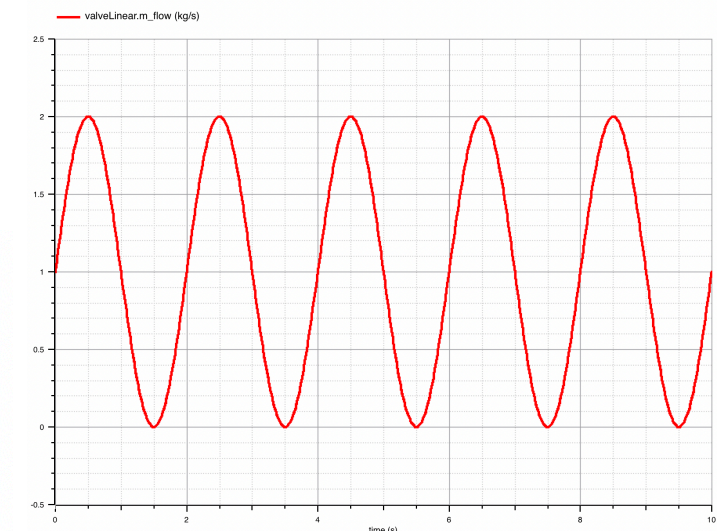
SimpleValve1 のソースコード

```
model SimpleValve1
  replaceable package Medium = Modelica.Media.Water.StandardWater;
  inner Modelica.Fluid.System system annotation( ...);
  Modelica.Fluid.Sources.Boundary_pT boundary(redeclare package Medium = Medium, T = 293.15, p = 121325, nPorts = 2)
    annotation( ...);
  Modelica.Fluid.Valves.ValveLinear valveLinear(redeclare package Medium = Medium, dp_nominal = 10000, m_flow_nominal = 1)
    annotation( ...);
  Modelica.Fluid.Sources.Boundary_pT boundary_pT(redeclare package Medium = Medium, T = 293.15, p = 101325, nPorts = 1)
    annotation( ...);
  Modelica.Blocks.Sources.Sine sine(amplitude = 0.5, freqHz = 0.5, offset = 0.5, phase = 0, startTime = 0) annotation( ...);
  Modelica.Fluid.Sources.Boundary_pT boundary2(redeclare package Medium = Medium, T = 293.15, nPorts = 1, p = 101325)
    annotation( ...);
  Modelica.Fluid.Valves.ValveDiscrete valveDiscrete(redeclare package Medium = Medium,
    dp_nominal = 10000, m_flow_nominal = 1, opening_min = 0.001) annotation( ...);
  Modelica.Blocks.Sources.BooleanPulse booleanPulse(period = 2, width = 50) annotation( ...);
equation
  connect(boundary.ports[1], valveLinear.port_a) annotation( ...);
  connect(valveLinear.port_b, boundary_pT.ports[1]) annotation( ...);
  connect(sine.y, valveLinear.opening) annotation( ...);
  connect(boundary.ports[2], valveDiscrete.port_a) annotation( ...);
  connect(valveDiscrete.port_b, boundary2.ports[1]) annotation( ...);
  connect(booleanPulse.y, valveDiscrete.open) annotation( ...);
  annotation( ...);
end SimpleValve1;
```

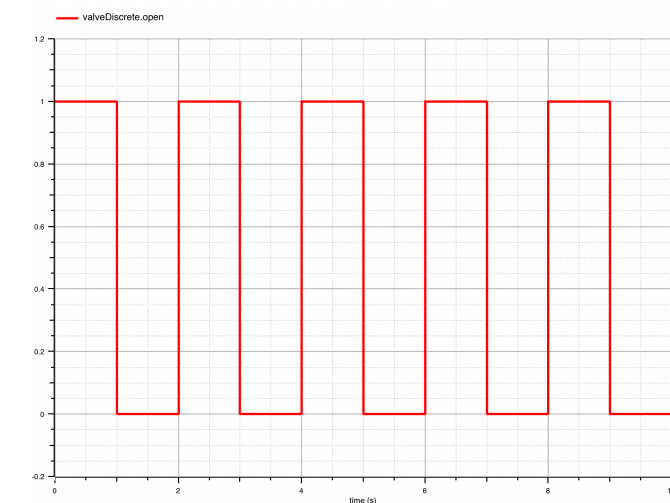
シミュレーション結果



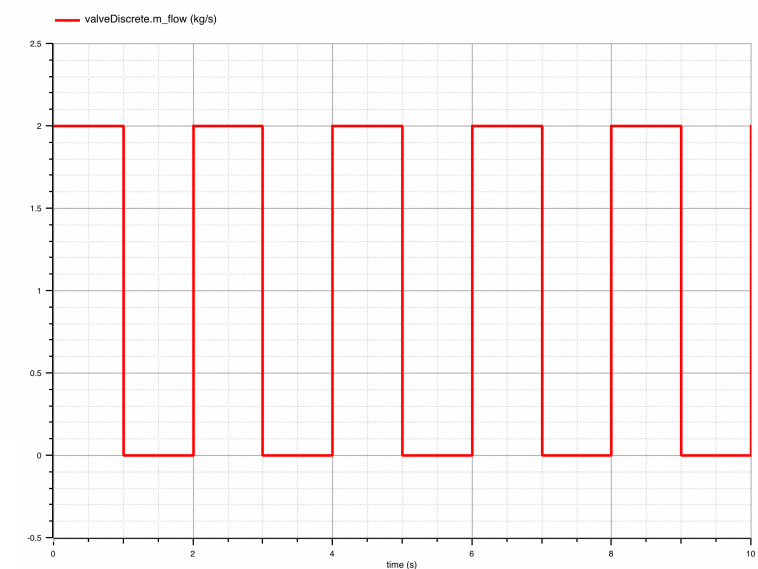
バルブ開度



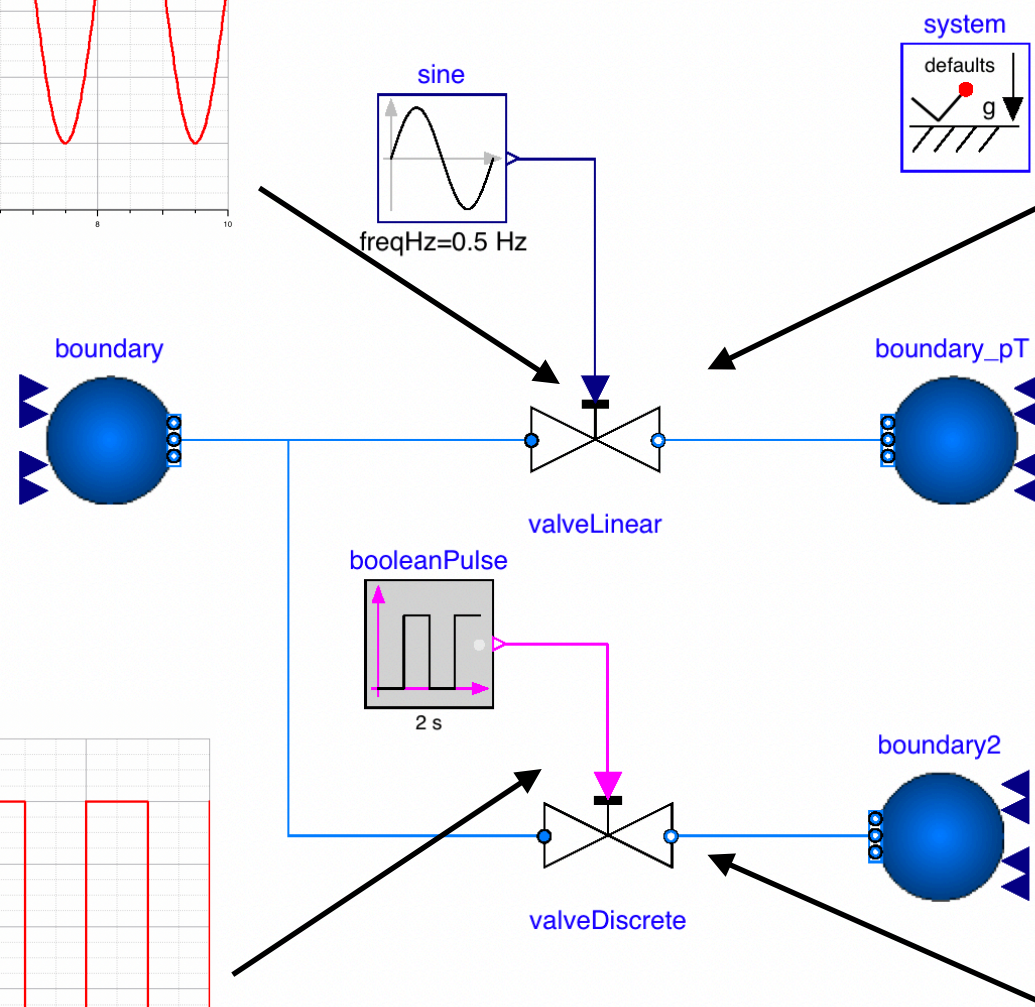
質量流量



バルブ開度



質量流量



まとめ

- ValveLinear と ValveDiscrete は、質量流量が圧力差 (dp) と開度 (opening) に比例するバルブモデルである。
- 流量特性は、全開時の基準流量 (m_flow_nominal) と基準圧力差 (dp_nominal) を設定する。
- バルブ通過時の発熱は考慮せず、エンタルピーは変化しない。
- ValveLinear は、開度(opening)としてRealInput コネクタ opening に 0 から 1 の実数値を設定する。
- ValveDiscrete は、開度として、BooleanInput コネクタに true または false を設定する。true のときは 開度 = 1, false のときは 開度 = opening_min となる。
- バルブ内の温度や密度は、上流側の値を使用する。
- 流れの向きが変化するときなどの質量流量が小さい領域 (m_flow_small 以下) 場合は、体積流量が不連続に変化しない適正化 (regularization) がなされる。

Licensed by Amane Tanaka under the Modelica License 2

Copyright(c) 2019, Amane Tanaka

- The purpose of this document is introducing the valve models of Modelica.Fluid which are included in the Modelica Standard Library (MSL). This document uses libraries, software, figures, and documents included in MSL, and those modifications. Licenses and copyrights of those are written in next page.
- This document is free and the use is completely at your own risk; it can be redistributed and/or modified under the terms of the Modelica license 2, see the license conditions (including the disclaimer of warranty) at <http://www.modelica.org/licenses/ModelicaLicense2>

Modelica Standard Library License

<https://github.com/modelica/ModelicaStandardLibrary/blob/master/LICENSE>

BSD 3-Clause License

Copyright (c) 1998-2019, Modelica Association and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.