WORKSHOP 3.2

Writing Modelica code

Contents

Introduction

The pump and tank system provided in the workshop library will be extended with a model for a solar panel. For reference, see image below. The solar collector will be attached to the pipe before the tank and used to heat up water, which is pumped through it before entering the tank. The solar panel model will be a combination of new Modelica code and existing sub-components from the Modelica Standard Library.
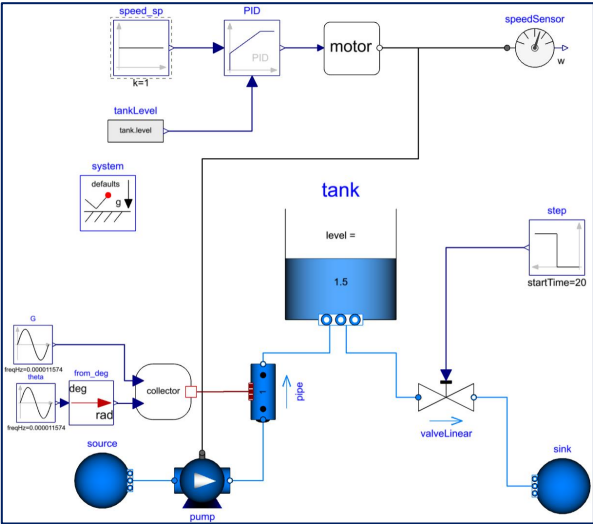
Solar panel model

Only the solar heat flow into the panel needs to be covered by the new model to be written, the water side including heat transfer is already present in the existing dynamic pipe model from the *Modelica.Fluid* library. The following mathematical model describes the heat transfer of a solar panel.

Mathematical model of a solar collector

$$\dot{Q} = \eta \cdot A \cdot G$$

where $\dot{Q}$ is the heat flow rate in W, η is the collector efficiency, A is the collector area and G is the global radiation incident.

The collector efficiency is determined as follows:

$$\eta = F \cdot \left(a_0 - a_1 \cdot \frac{\Delta T}{\max{(G, 0.0001)}} - a_2 \cdot \frac{\Delta T^2}{\max{(G, 0.0001)}} \right)$$

where a_0, a_1 and a_2 are collector specific characteristic parameters, ΔT is the temperature difference between fluid (average) and environment:

$$\Delta T = T_{\text{m}} - T_{\text{amb}}$$

The ambient temperature T_{amb} is assumed to be constant at 15 °C. The F factor takes into account the effect of the angle incidence θ

$$F = \begin{cases} 1 - b_0 \cdot \left(\dfrac{1}{\cos\theta} - 1 \right) & \text{for } \theta \leq \theta_0 \\ 0 & \text{for } \theta > \theta_0 \end{cases}$$

The characteristic parameters for the collector to be used in this example are:

a_0	a_1	a_2	b_0	θ_0	A
0.75	4.4	0.004	0.108	84.4° = 1.473 rad	500 m²

The angle of incidence θ and the global radiation incident G are determined outside this component and should be modeled as input signals.

Modeling
1. Go to the package **Day3.W2** and create a new model class called **SolarCollector**
2. Open the icon editor and draw a simple icon.
3. Open this new **SolarCollector** model and drag in connectors for variables that cross the component boundary.
   a. Input connectors for variables (G and theta) that are input signals:
   Modelica.Blocks.Interfaces.RealInput. This connector is a bit special. Its name (changed by double-clicking on it and selecting the name in the right panel) can be used directly as the input variable, i.e. no dot-notation is required.
   b. Thermal connector for heat exchange with water. It contains the water temperature and the heat flow rate: ***Modelica.Thermal.HeatTransfer.Interfaces.HeatPort_b.*** Important: Flows that leave the component are defined as negative values.
4. Open the code editor and declare all parameters of the solar collector, i.e., all variables that do not vary with time.
5. Declare all remaining variables, which are required to describe the problem.
6. Add the model equations as defined above, use an **if-statement** for the F-factor correlation.

Unit issues:

- To ensure quantity compatibility and benefit from preferred display units, all parameters and variables should be declared in SI-units. Use predefined types from the package *Modelica.SIunits*.
- Observe also that trigonometric functions, like cos, expect arguments in rad, which may be converted first (π can be found in *Modelica.Constants*).
- *Modelica.SIunits.Temperature* is unsuitable for temperature differences, since its display unit is *degC*, which means that 273.15 is subtracted from its value for plotting. Use *Modelica.SIunits.TemperatureDifference* instead.
- Display units are only used for plotting and in input dialogs, they do not influence model equations.
- Clicking on the **Packages** bar in the package browser sorts all classes in alphabetical order, which makes it easier to find a certain type in the *Modelica.SIunits* package.

Create test model

1. When the component is implemented and a check of the new model is ok, a test model for the new component should be created. Create a new model named **SolarCollectorTest** in package **Day3.W2**.
2. Drag in a component of your solar collector model and set its parameters given in the Modeling section.
3. Drag in a component of the class *Modelica.Blocks.Sources.Sine*, which is used as a source block for the angle of incidence theta and parameterize it as follows:

FreqHz	amplitude	phase	Offset
1/86400	60	90 deg	100

Note, theta in *SolarCollector* should be declared as radians. We therefore need to transform the output of the theta sine source to radians. This is done by dragging in component *Modelica.Blocks.Math.UnitConversion.From_deg*, see Figure 1.

4. Drag in a component of the class *Modelica.Blocks.Sources.Sine*, which is used as a source block for the global radiation incident G and parameterize it as follows:

FreqHz	amplitude	phase
1/86400	1000	-90 deg

5. Drag in component of the class *Modelica.Thermal.HeatTransfer.Sources.FixedTemperature* and set the parameter T to 15 *degC*. The system should now look like the Figure 1.
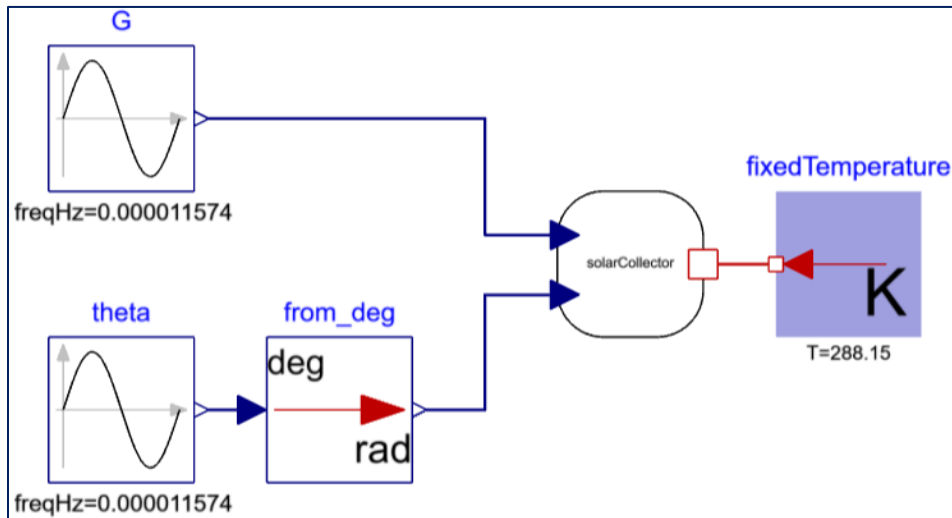
Figure 1 New layout of pump system

6. Simulate the model for 86400 seconds and plot the heat flow in the heat connector and the *F* factor. If implemented correctly, the results should look like Figure 2.
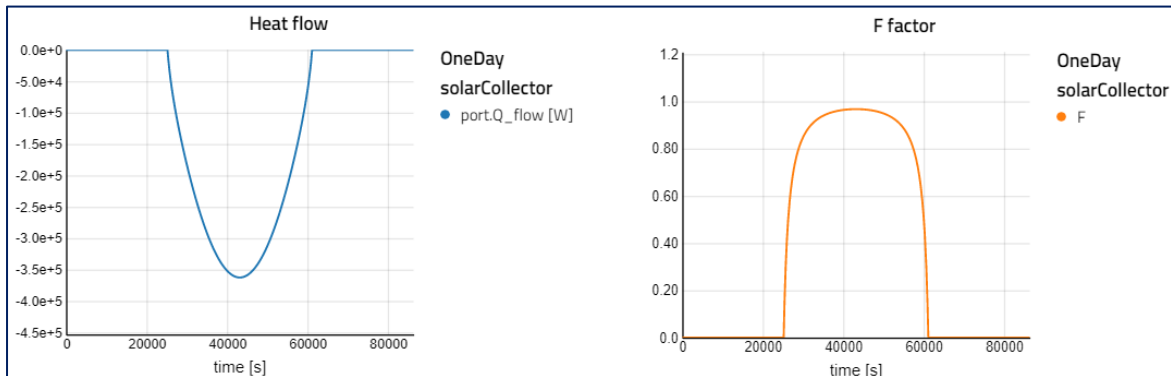

Figure 2 SolarCollector Test

Integrate collector in system model

1. Open the system model **PumpSystemIII** located in the **Day3.W2** package.
2. Change the pipe model class from **StaticPipe** to **DynamicPipe** from the same package, located at *Modelica.Fluid.Pipes.DynamicPipe*.
3. Change the following settings in the **pipe** component. Double-click to bring up the parameter dialog.

Where?	Assumptions	Assumptions	HeatTransfer edit button	Advanced	Advanced
What?	use_HeatTransfer	HeatTransfer	alpha0	nNodes	modelStructure
Value	true	Constant heat transfer	2000	1	av_b

Setting the parameter **nNodes** to 1 makes the pipe a lumped volume with one temperature which is accessed via the component thermal connector.

4. Drag in a component of your solar collector model the blocks used in model **SolarCollectorTest** **but** remove the temperature source and instead connect the *SolarCollector* to the pipe model.

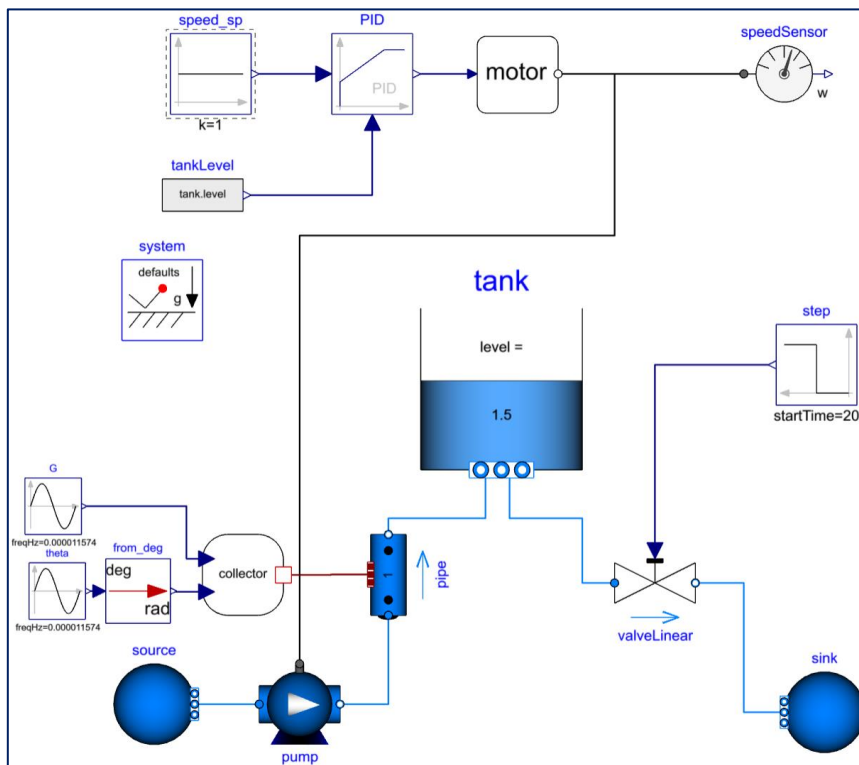The system should now resemble that in Figure 3.

Figure 3 New layout of pump system

Change parameter settings

5. In order to reduce the mass flow rate through the tank choose a different valve (use same model, change parameter *m_flow_nominal* to 5 kg/s) and open the valve only by 50% (*step.height*=0.5).

Simulate

6. Simulate the system for 24 *h* = 86400 s
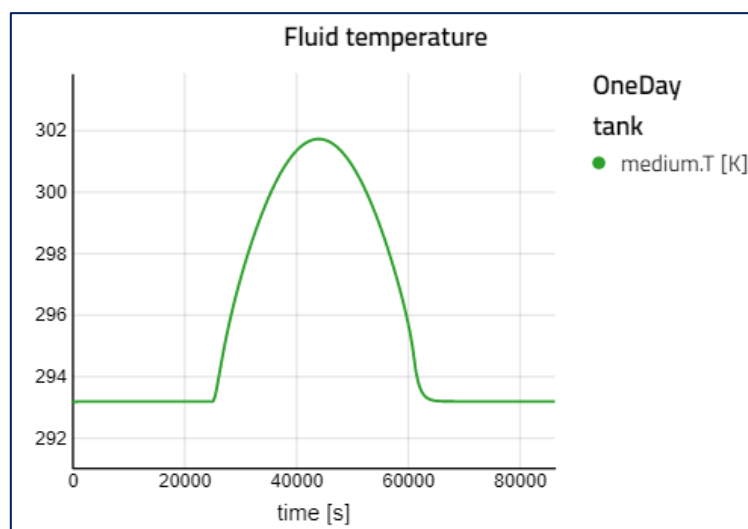7. What is the maximum temperature in the tank?

Compare your results with those plotted in Figure 4.

Figure 4 Tank temperature