# Generation of thermohydraulics condenser model

## Creation of a new TPP diagram

### New TPP diagram

Generate a new TPP project (design), “**New project**”→“**TPP diagram**” (see Figure 1). Using a standard “save file” dialog save the diagram under a new name in a newly created catalog: “C:\KTZ\Turbine\Condenser\Condenser KP-3200.prt” (create the catalog in advance).

### Global parameters

Let us first set global parameters (signals) for the condenser model. Go to **“Graphics”** → “Signals…” item via the main SimInTech menu.

In the window displayed we shall set four new signals – steam flow from the turbine, cooling water flow, second heat exchanger cooling water flow and temperature of cooling water supplied to condenser heat exchangers: “Gp”, “Gov1”, “Gov2” and “Tov”. Set their values as per Figure 33:

Gp – Steam flow from turbine – Real – Input – 125.0

Gov1 – Cooling water flow – Real – Input – 2500.0

Gov2 – Cooling water flow – Real – Input – 2500.0

Tov2 – Cooling water temperature – Real – Input – 10.1

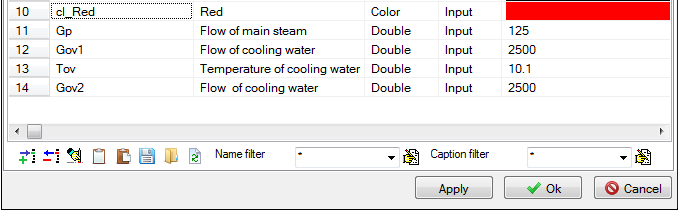


Figure 33. Global parameters for condenser model

We shall change these parameters in the process of diagram debugging; thus, create buttons (8 pcs.) on the diagram to change the parameters. To accelerate the process copy the buttons from the flowing part diagram and edit their properties, see Figure 34. It can be seen from the figure that the values have remained from the STP flowing part. The variables will be re-calculated on the first start-up.

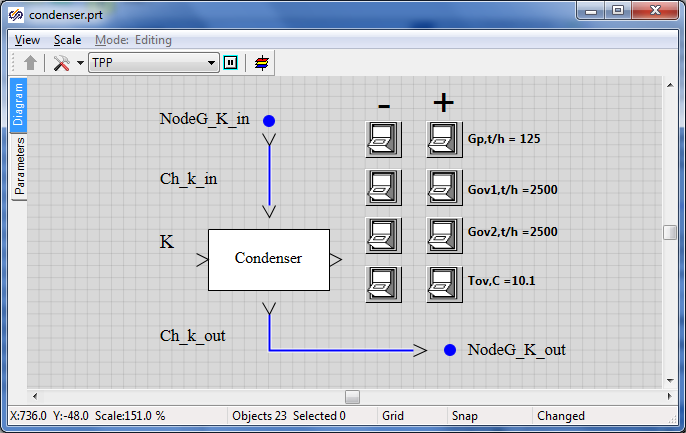


Figure 35. Buttons for changing global parameters in the condenser model

## Main condenser model

### Model description

In this tutorial example we shall create the condenser model using 3-D TPP tank (compensator). Waste steam from turbine LPC will be supplied to the top volume of the tank and condensed in the tank. 2 heat exchangers will be connected to the tank to remove heat from the condenser. Condensate from the bottom portion of the tank will be removed to boundary conditions.

Along with creation of a condenser we will learn how to use SimInTech submodel for creation of new blocks (diagram elements) with setting properties in those. We will create a condenser with an option to change such properties as steam space volume, heat transfer surface, number of cooling tubes, etc., see Figure 35. Then the debugged and verified substructure can be easily (simply by copying) transferred to other projects along with entering, as a rule, minor corrections. It considerably reduces time for development and debugging of new diagrams.

There will be two boundary conditions, both of them of G-type (in nominal stationary condition – the amount of steam delivered from the turbine to the condenser will be the same as the amount of condensate drained out).

Cooling water lines will be modeled in the simplest way – by means of common-mode channels between boundary nodes of G and P-types and heat port.

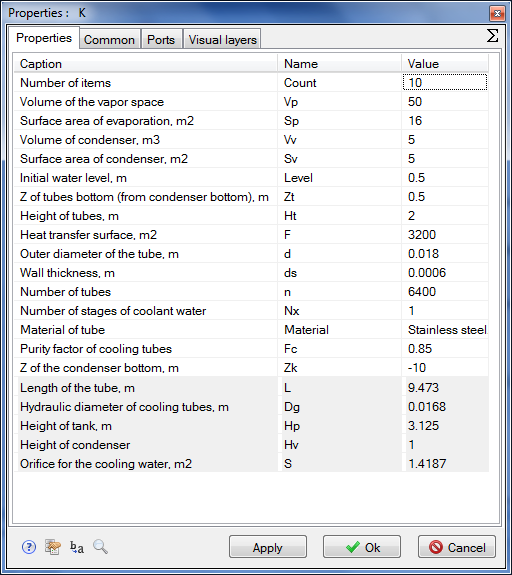


Figure 35. Properties of condenser submodel

### Creation of condenser model top level

So, let us consequently arrange the following elements in the diagram window and rename those for the sake of convenience.

1. **“Boundary node G”**, object name: **“NodeG\_K\_in”**
2. **“Boundary node G”**, object name: **“NodeG\_K\_out”**
3. **“SimInTech submodel”**, object name: **“K”**, element type (ClassName): **“TPP Condenser”** (instead of **“SimInTech submodel”**)
4. **“Common-mode channel”**, object name: **“Ch\_K\_in”**
5. **“Common-mode channel”**, object name: **“Ch\_K\_out”**

Using mechanism for displaying parameters of the element of the diagram, enter the names of newly set elements in the diagram window. Go to the condenser properties and change the picture by default for **“Condenser KP-3200”** caption; meanwhile, align (bind) the caption to the center and to the middle (by the width and the height). Then place the caption in the center of graphical editor window, for which purpose:

1. Go to **“Object properties” → “Graphical image”**. Erase the image there and insert the text using the primitives panel.
2. Then go to the text properties and set **“Insertion point position”** and **“Alignment style”** to **“On center”** position. Press **“OK”**.
3. Place the text so that the alignment point coincides with the center of submodel rectangular image and scale up to **“Fit frame”**. Align the text straight to the center, close the window after saving changes in the image. Make sure that the submodel view in the diagram window has been changed.
4. Widen the area of the submodel block to ensure visibility of the caption by pulling the block with its right bottom corner.
5. Double-click to enter inside the submodel and place the two blocks there: **“TPP input port”** and **“TPP output port”**.
6. Exit the submodel to the top level (by double-clicking the free area of the diagram window). The diagram shall appear similar to Figure 36:

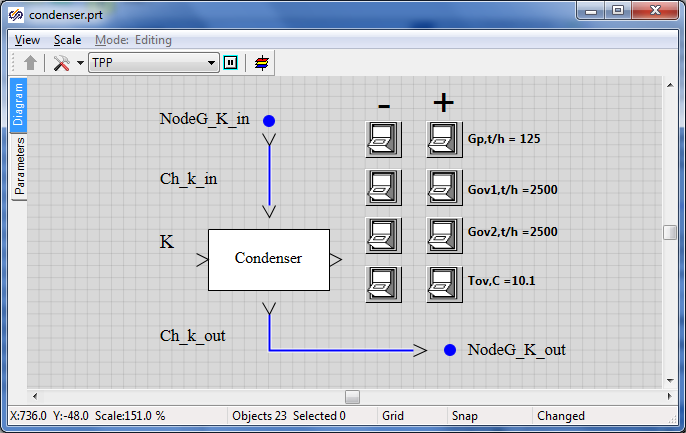


Figure 36. Creation of condenser model (beginning)

1. Change the position of submodel ports: input port – **“On top”**, output block – **“On bottom”**.
2. Link up all elements on the diagram with trace lines.
3. Change the color of **“Ch\_K\_in”** channel to red or orange to highlight the steam flow via this channel.
4. Place the arrow next to the boundary node so that it is to the right from the node.
5. Compare the result with Figure 37.

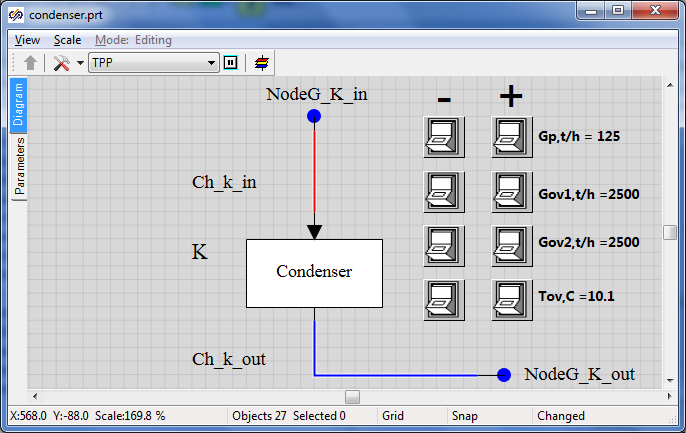


Figure 37. Creation of condenser model (top level is ready)

### Creation of condenser submodel nested level

Now go to setting of the thermohydraulics diagram inside the submodel:

1. Enter into the submodel.
2. Rename the input port as **“LPC exhaust”**.
3. Rename the output port as “**Output of condensate**”, see Figure 38.

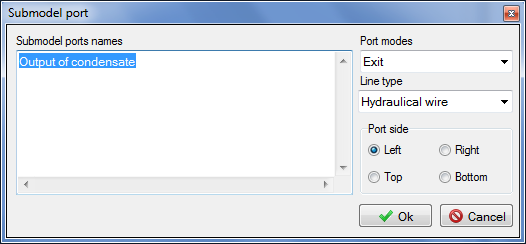


Figure 38. Change of output port name

1. Place two **“Boundary nodes G”** at the left.
2. Place two **“Boundary nodes P”** at the right on the diagram.
3. Place two **“common-mode channels”** on the diagram (between boundary nodes; it will be a model of heat exchanger tubing).
4. Place **“3V compensator”**. Increase the block size.
5. Place another **“Compensator node”** on the compensator on its top.
6. Place two **“Local resistance”** elements (from **“Valves”** tab) in each channel: in the channel head and in the channel end.
7. Link up **“LPC exhaust”** input port and **“Output of condensate”** output port with internal condenser nodes (3V TPP compensator). Please note that we are linking up not by means of channels but just by hydraulic connections. Common-mode channels used for supplying steam and draining condensate have been created earlier, outside the submodel.
8. Change the color of steam supply lines and the color of correspondent condenser node for orange.
9. Change the color of condensate drain line and condenser node for blue.

Compare the result with Figure 39.

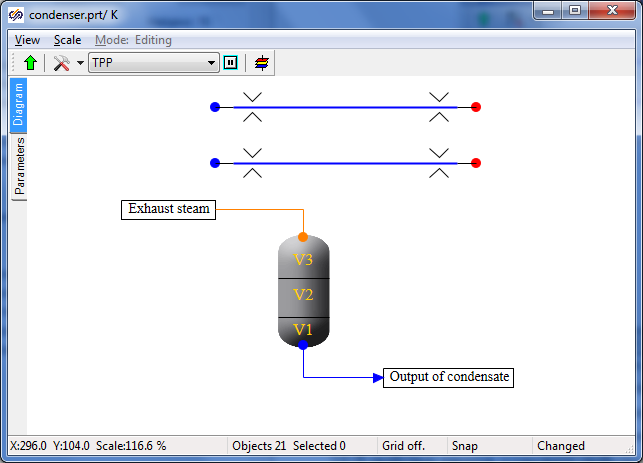


Figure 39. Condenser submodel (nested level)

Now we only need to link up heat exchanger lines with heat links to the condenser and rename objects as convenient for further programming.

1. Change the number of heat ports in the condenser properties for **“2”**.
2. Set **“Heat link”** property in the channel properties to **“Yes”**.
3. Link up heat exchanger channels to condenser heat ports.
4. For convenience, set “**Yes**” in the “**Show frame**” property for the condenser. The result shall be as depicted in Figure 40.

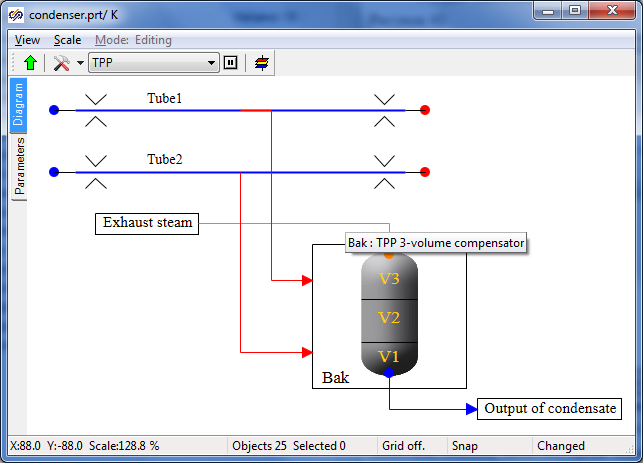


Figure 40. Condenser submodel with heat links (nested level)

1. Rename the channel of the first (top) heat exchanger as **“Tube1”**, the channel of the second (bottom) heat exchanger as **“Tube2”** (the **“Object name”** property in the object properties).
2. Rename the 3V compensator element as **“Bak”**.
3. Display the element names on the diagram, see Figure 41.

Now the setting of the diagram is completed (for now), here we shall set object parameters and, in principle, it can be done in a similar way as we have set those for flowing part but we will do it in a more universal, program manner using the new block editor and the SimInTech built-in programming language.

### New block editor

A submodel that will be then conveniently transferred to other projects and used there shall be fully determined from the top level in parametric representation. I.e., it will be sufficient to change the heat exchange surface in the properties of the submodel itself. While all properties of all elements dependant on the heat exchange surface will be automatically re-calculated for new values inside that. It will be done in the same way as for other parameters.

Go to the top level of the diagram window and select the submodel properties. Make sure that the object name has been set: **“K”** (English letter); element type (ClassName) has been set: **“TPP Condenser”** (instead of **“SimInTech submodel”**). **Class name setting is a matter of primary** importance since all further manipulations SHALL NOT disturb and change the TPP and SimInTech standard library.

Close the properties panel and, on selecting the submodel in the diagram window (by mouse single-clicking), go to the SimInTehch main menu, item **“Editing”** → “Change block…”. It will open the dialog window with tabular settings of parameters and properties of the block selected on the diagram (i.e., condenser submodel).

Here we will need to set all properties, their names and values determining the condenser model. Totally 21 properties shall be added, from which number 5 will be reference ones (non-editable but unambiguously calculated from the previous parameters).

1. Create 21 strings and carefully and successively set the caption, name, data type, value and calculation method for each property, see table 3.1.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table 3.1 | | | | | |
| **No.** | **Description** | **Name** | **Data type** | **Value** | **Calculation method** |
| 1 | Number of elements | Count | Integral | 10 | Constant |
| 2 | Steam space volume | Vp | Real | 50 | Variable |
| 3 | Surface area of evaporation, m2 | Sp | Real | 16 | Variable |
| 4 | Volume of condenser, m3 | Vv | Real | 5 | Variable |
| 5 | Surface area of condenser, m2 | Sv | Real | 5 | Variable |
| 6 | Initial water level, m | Level | Real | 0.5 | Variable |
| 7 | Z of tubes bottom from condenser, m | Zt | Real | 0.5 | Variable |
| 8 | Height of tubes, m | Ht | Real | 2 | Variable |
| 9 | Heat transfer surface, m2 | F | Real | 3200 | Variable |
| 10 | Outer diameter of the tube, m | d | Real | 0.018 | Constant |
| 11 | Wall thickness, m | ds | Real | 0.0006 | Constant |
| 12 | Number of tubes | n | Integral | 6400 | Constant |
| 13 | Number of stages of cooling water | Nx | Real | 1 | Variable |
| 14 | Material of tubes | Material | Database file name | 18ХН9Т | Constant |
| 15 | Purity factor of cooling tubes | Fc | Real | 0.85 | Variable |
| 16 | Z of the condenser bottom, m | Zk | Real | -10 | Variable |
| 17 | Length of the tube, m | L | Real | 9.4735 | Constant |
| 18 | Hydraulic diameter of cooling tube, m | Dg | Real | 0.0168 | Variable |
| 19 | Height of tank, m | Hp | Real | 3.125 | Variable |
| 20 | Height of condenser | Hv | Real | 1 | Variable |
| 21 | Flow area for the cooling water, m2 | S | Real | 1.4186 | Variable |

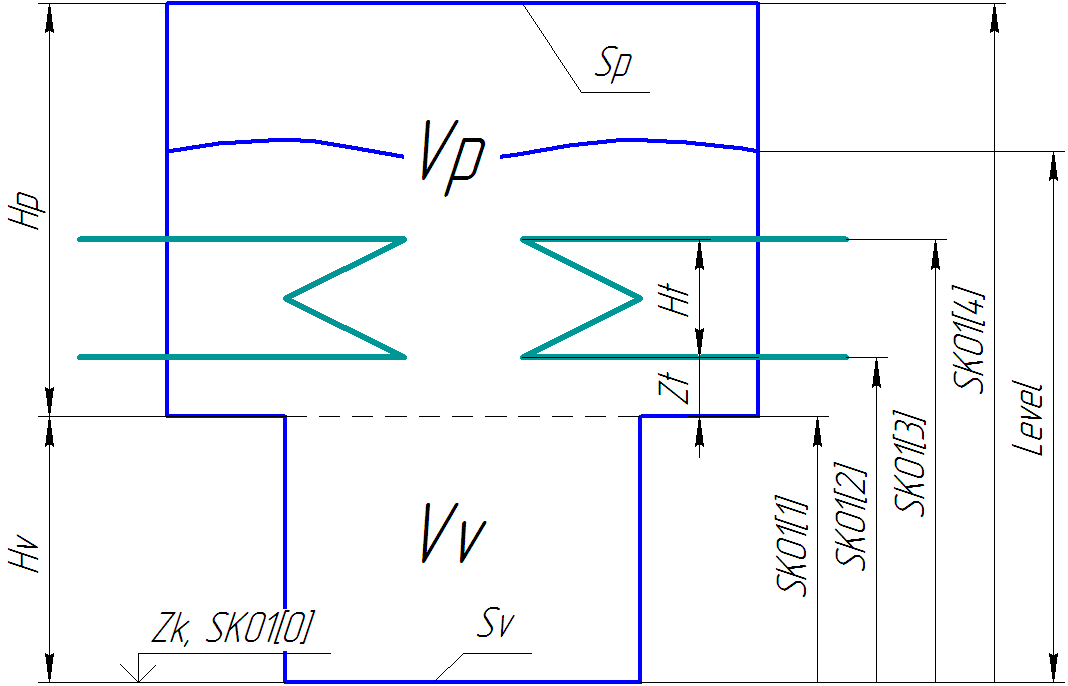


Figure 41. Condenser geometrical model

1. After completion of all properties, compare the result with Figure 43 and re-check all entered fields and values.
2. Now, on pressing “**Ok**” and returning to the diagram, you can try to go to the submodel properties and see all entered parameters in “Properties” tab, see Figure 43.
3. As it was said before, not all of these properties shall be editable, i.e., the user will need to set all those by hand. Five properties shall be “read only”. Again go to the new block editor, on previously selecting the submodel. Write down read-only properties separated by semicolon in “**Read-only properties**” at the right bottom part of the editor: “**L;Dg;S;Hp;Hv**”. See example in Figure 43.
4. Press **“OK”**. Go to the submodel properties: now the five strings are highlighted grey: it means that these properties shall not (i.e., meaninglessly) be changed by hand since those will be recalculated at the initialization stage or at the very first step of integration (the algorithm for their calculation can be found below).

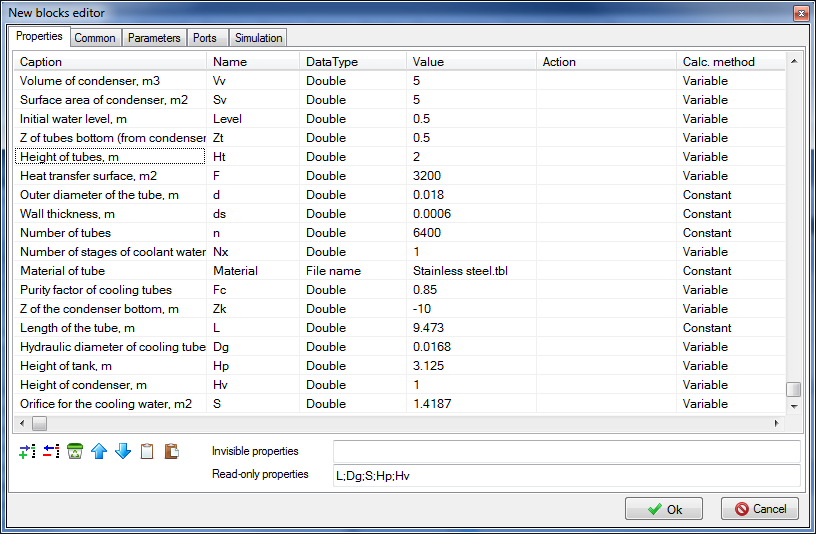


Figure 42. New block editor (condenser)

Now let us find out how the properties can be initialized in the submodel.

1. Go to the nested level of the submodel. Open “Parameters” tab there (at the left, under “Diagram” tab). An empty submodel global parameter editor window will be opened. Whatever you want, can

be written here in the programming language. Start with setting the following strings in this window:

|  |
| --- |
| **initialization**  submodel.Dg = submodel.d - 2\*submodel.ds;  **setpropevalstring**(submodel,"Dg",submodel.Dg);  submodel.S = pi\*submodel.Dg\*submodel.Dg\*n/4;  **setpropevalstring**(submodel,"S",submodel.S);  **setpropevalstring**(submodel,"L",submodel.F/(pi\*submodel.Dg\*submodel.n));  **end;** |

Here, in the initialization block, we calculate the hydraulic diameter of tubes (outer diameter minus double thickness of the wall). Then prescribe an obtained value in the submodel properties. After that calculate the flow area for cooling water (the product of the section area of one tube by the number of tubes) and length of tubes (area of heat exchange surface shall be divided by the length of inner section circle of one tube and by the number of tubes).

Note that properties can be calculated by both a separate string and inside the function call  
**“setpropevalstring()”**. See also Figure 44.

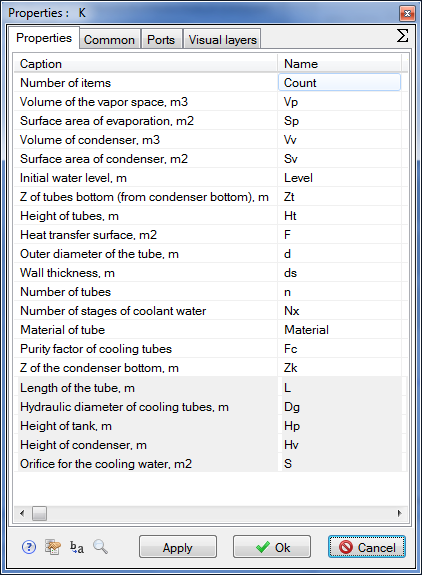


Figure 43. New submodel (condenser) properties

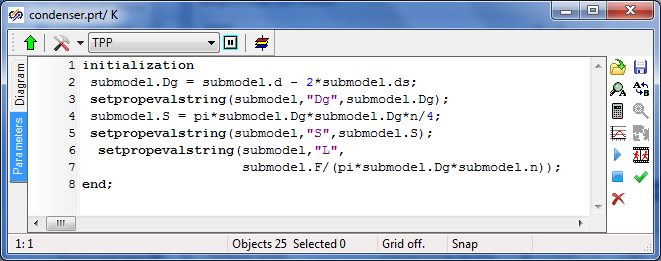


Figure 44. Condenser submodel parameters editor (calculation of properties)

1. After that press the tick in the menu at the right and, on navigating on the diagram window, save the project. Then press “**Initialization**” button (clock sign in the SimInTech top toolbar, Figure 45). Then, after initialization of the diagram, “**Stop**” red button close to the initialization button can be pressed.

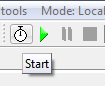


Figure 45. Calculation initialization button

1. Now again go to the submodel properties and make sure that three properties have been re-calculated and a new value has been assigned to those, see Figure 47 (re-calculated properties can be seen due to a large number of numerals after the dot):

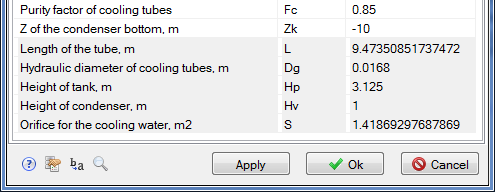


Figure 46. Recalculation of submodel properties

1. Let us move farther, complete the following strings in the submodel global parameters editor (as well as inside the initialization block in addition to the already written):

|  |
| --- |
| submodel.Hp = submodel.Vp/submodel.Sp;  **setpropevalstring**(submodel,"Hp",submodel.Hp);  submodel.Hv = submodel.Vv/submodel.Sv;  **setpropevalstring**(submodel,"Hv",submodel.Hv);  **if** submodel.Level > submodel.Hp+submodel.Hv  **then** submodel.Level = submodel.Hp+submodel.Hv;  **if** submodel.Zt > submodel.Hp  **then** submodel.Zt = submodel.Hp;  **if** (submodel.Zt+submodel.Ht) > submodel.Hp  **then** submodel.Ht = submodel.Hp-submodel.Zt;  **setpropevalstring**(submodel,"Zt",submodel.Zt);  **setpropevalstring**(submodel,"Ht",submodel.Ht);  **setpropevalstring**(submodel,"Level",submodel.Level); |

First we will calculate the tank height for steam and water and assign the calculated value to particular peroperties. Then we will carry out checks:

* initial water level shall not be higher than the height of the condenser (total heights of steam and condensate parts);
* Z of tubes bottom shall not be higher than the height of steam space;
* Z of tubes top shall not be higher than the height of steam space, otherwise the height of tubes shall be reduced.

After checking, prescribe the three checked properties.

1. Now let us go to the essence – to the setting of properties of objects, which are inside the submodel. It is done in the same way as described before – just specify what object is assigned with one or another property. Add to previous strings the following ones, in which we will calculate height level distribution (SKO2), relative heat exchange areas for water (SKO3) and steam (SKO4) depending on the filled volume (SKO1) for **“3V compensator”** object named as **“Bak”**.

|  |
| --- |
| sko1 = [0,  submodel.Vv,  submodel.Vv+submodel.Sp\*submodel.Zt,  submodel.Vv+submodel.Sp\*(submodel.Zt+submodel.Ht),  submodel.Vv+submodel.Vp];  **setpropevalstring**(Bak,"SKO1","["+sko1+"]");  sko2 = [0,  submodel.Hv,  submodel.Hv+submodel.Zt,  submodel.Hv+(submodel.Zt+submodel.Ht),  submodel.Hv+submodel.Hp];  **setpropevalstring**(Bak,"SKO2","["+sko2+"]");  sko3=[0, 0, 0, 1, 1];  **setpropevalstring**(Bak,"SKO3","["+sko3+"]");  sko4=[1, 1, 1, 0, 0];  **setpropevalstring**(Bak,"SKO4","["+sko4+"]");  tmp = min(submodel.Level,submodel.Hv)\*submodel.Sv  +  max(0,submodel.Level-submodel.Hv)\*submodel.Sp;  **setpropevalstring**(Bak,"V1",0.2\*tmp);  **setpropevalstring**(Bak,"V2",0.8\*tmp);  **setpropevalstring**(Bak,"V3",submodel.Vv+submodel.Vp-tmp);    InitObject(Bak); |

Here in sko1, sko2, sko3 and sko4 arrays we will use points to set numerical values, which will be used by TPP code to build piecewise-linear dependences for the level in condenser and relative heat exchange areas for water and steam on the filled condenser volume.

Then calculated arrays will be entered into related properties of **“Bak”** object.

In **“tmp”** variable we will calculate the condenser volume filled with water (liquid phase) in the initial point of time and in dependence with that calculate three condenser volumes **“V1”**, **“V2”** and **“V3”**.

**“**InitObject(Bak)”function forcedly initiates and renews values of **“Bak”** properties.

Then the diagram can be re-initialized and properties in “**Bak**” object can be viewed: those shall have been recalculated, see Figure 47, fields “SKO1”, “SKO2”, “SKO3”, “SKO4” and “V1”, “V2”, “V3”.

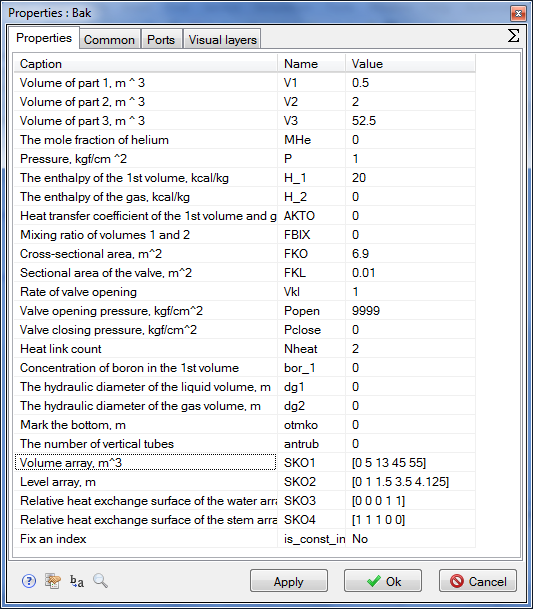


Figure 47. Recalculation of “Bak” object properties (3V compensator)

1. Let us go to the setting of properties for heat exchanger tubes. Each channel will consist of 10 sections (submodel **“Count”** parameter) in terms of the length. Then hydraulic diameter; flow area; length; hydraulic resistance; reverse hydraulic resistance; X, Y, Z increment; initial boron concentration; wall thickness; heat exchange surface area; degree of roughness; number of heat link elements shall be set for each section. It is executed in a rather simple manner in the programming language.

Set the following text in the initialization block:

|  |
| --- |
| Tube1.Material=Material;  **setpropevalstring**(Tube1,"Count" ,submodel.Count);  **setpropevalstring**(Tube1,"Gidr\_D" ,"["+Count#submodel.Dg +"]");  **setpropevalstring**(Tube1,"Sechen" ,"["+Count#submodel.S/2 +"]");  **setpropevalstring**(Tube1,"Dlina" ,"["+Count#(submodel.L/Count)+"]");  **setpropevalstring**(Tube1,"Soprot" ,"["+Count#2/Count +"]");  **setpropevalstring**(Tube1,"InvSopr" ,"["+Count#2/Count +"]");  **setpropevalstring**(Tube1,"Z" ,"["+Count#0 +"]");  **setpropevalstring**(Tube1,"X" ,"["+Count#0 +"]");  **setpropevalstring**(Tube1,"Y" ,"["+Count#0 +"]");  **setpropevalstring**(Tube1,"Bor" ,"["+Count#0 +"]");  **setpropevalstring**(Tube1,"Sten" ,"["+Count#(submodel.ds/submodel.Fc) +"]");  **setpropevalstring**(Tube1,"F" ,"["+Count#(submodel.F/Count/2) +"]");  **setpropevalstring**(Tube1,"Rz1" ,"["+Count#0 +"]");  **setpropevalstring**(Tube1,"HeatElements","["+Count#3 +"]");  InitObject(Tube1); |

On pressing “**OK**” and initializing the diagram, one can make sure that all properties have been set in “**Tube1**” channel exactly as it has been just programmed, see Figure 48:

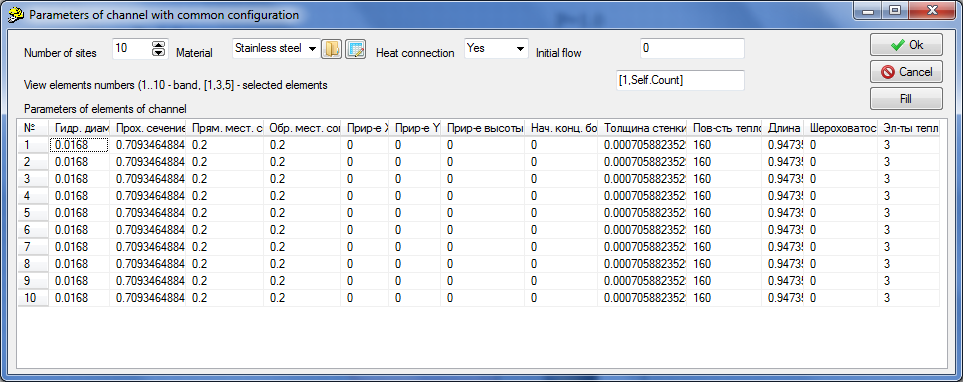


Figure 48. Recalculation of “Tube1” object properties (common-mode channel)

Note that channel properties will be automatically recalculated if we change numerical values of one or another submodel property.

1. Now for the purpose of training set all properties for “**Tube2**” channel independently and by analogy using the initialization block. Then make sure that everything works properly and “Tube2” channel properties are the same as depicted in Figure 48.
2. Then, the only thing remained is to set a mechanism in the condenser submodel to dynamically display parameters needed for us on the screen in the process of calculation. To that end place the following code AFTER the initialization block:

|  |
| --- |
| submodel.\_G = -(Tube1.G+Tube2.G)\*3.6;  submodel.\_w = -2\*Tube1.q[1]/submodel.S;  submodel.\_Tin = Tube1.\_Tvh;  submodel.\_Tou = Tube1.\_Tvyh;  submodel.\_dPtr = abs(Tube1.\_Pvh-Tube1.\_Pvyh);  submodel.\_Q = (Tube1.\_Qto+Tube2.\_Qto)\*4.182e-3;  submodel.\_Qf = submodel.\_Q/submodel.F;  submodel.\_dTou = Bak.Tpar\_-Tube1.\_Tvyh;    submodel.\_Level= Bak.L;  submodel.\_Ts = Bak.Tpar\_;  submodel.\_Ps = Bak.P\_; |

It is seen from the code that the following are calculated: cooling water mass flow; cooling water speed; input inlet and outlet cooling water temperature; pressure loss on tubes; heat flow (conversion from kkal to kW); specific heat flow; difference between tubes outlet water temperature and steam temperature; level in condenser; steam temperature and pressure in condenser.

1. To make these (and some other) parameters visible on the top level we need to go to the new block editor in the submodel parameters (on selecting the condenser submodel in the diagram window on the top level, in advance) and go to the **“Parameters”** tab. Here we will set next 16 parameters, see table 3.2 (for checking see Figure 49). Initial values of all parameters are zero, mode is “Output”.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Table 3.2 | | | | | |
| **No.** | **Caption** | **Name** | **Data type** | **Duty** | **Calculation method** |
| 1 | Steam flow to condenser, t/h | \_Gsteam | Real | Output | Variable |
| 2 | Steam enthalpy in condenser, kkal/kg | \_Hsteam | Real | Output | Variable |
| 3 | Pressure in condenser, atm | \_Ps | Real | Output | Variable |
| 4 | Temperature in condenser, C | \_Ts | Real | Output | Variable |
| 5 | Condensate enthalpy, kkal/kg | \_Hs | Real | Output | Variable |
| 6 | Cooling water flow, t/h | \_G | Real | Output | Variable |
| 7 | Hydraulic resistance for cooling water, kgf/cm2 | \_dPtr | Real | Output | Variable |
| 8 | Cooling water speed in tubes, m/s | \_w | Real | Output | Variable |
| 9 | Inlet cooling water temperature, C | \_Tin | Real | Output | Variable |
| 10 | Outlet cooling water temperature, C | \_Tou | Real | Output | Variable |
| 11 | Average heat transfer factor, W/m2\*K | \_alfa | Real | Output | Variable |
| 12 | Temperature difference at condenser outlet, C | \_dTou | Real | Output | Variable |
| 13 | Log mean temperature difference, C | \_LMTD | Real | Output | Variable |
| 14 | Total heat load, MW | \_Q | Real | Output | Variable |
| 15 | Specific heat load, kW/m2 | \_Qf | Real | Output | Variable |
| 16 | Water level, m | \_Level | Real | Output | Variable |

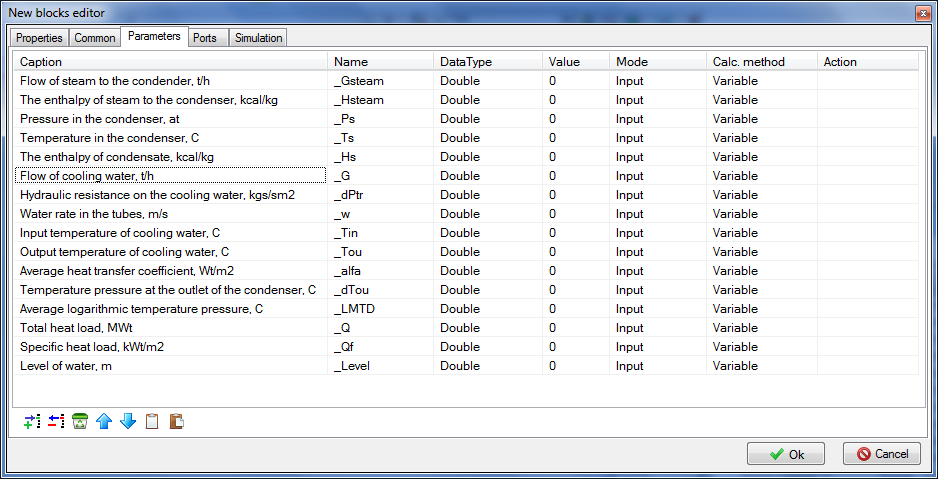


Figure 49. Condenser submodel parameters

1. Now, after saving the entered table in the condenser substructure model, you can go to “**Parameters**” tab (from the diagram window by right-clicking the condenser submodel) and set the list of entered parameters, which can be displayed on the graph either as a text or as a diagram window, see Figure 50.

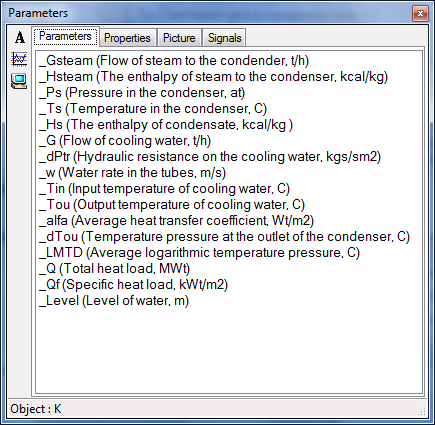


Figure 50. Display of set condenser submodel parameters

Thus, we have created the condenser model, which shall not be entered into to set the properties but properties may (shall) be set from the top level and those will be recalculated according to a written algorithm and transmitted to related elements inside the condenser.

Further then, for example, when the condenser model is completed, become more universal, fully debugged and run in such a submodel may be integrated into the elements pallet and be repeatedly used in many projects as an individual block.

### Display of parameters in diagram window

Let us display those parameters that will be interesting for us in the process of debugging of the condenser model in the diagram window.

1. For all channels (that are only four on this diagram) display related flows in tons per hour (to that end it is not necessary to display parameter **“g”** itself, it shall be just multiplied by 3600/1000, i.e. **“g\*3.6”**.
2. For all boundary nodes (four nodes of G-type and two nodes of P-type) – display pressure, enthalpy and temperature in those (it is more convenient to do that using **“P, H, T control in the node”**).
3. For the condenser model – display temperature, pressure and water level in that: P, T, L. To display the level in mm multiply the condenser parameter in the settings of caption properties by 1000: “**\_Level\*1000**”, see Figure 52. Also carefully and attentively watch the text of all captions and measurement units for all values.

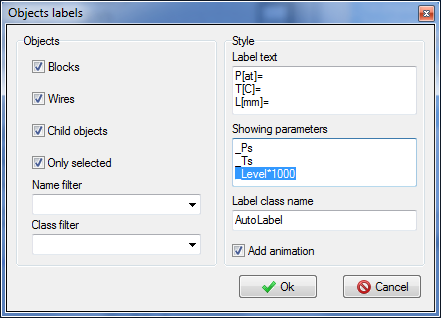


Figure 52. Display of condenser parameters P, T, L in the diagram window

1. For the condenser model – also display the following parameters useful for debugging: G[t/h], w[m/s], Tin[С], Tou[С], dT[C], dP[kgf], Q[MW], Qf[kW/m2], the result can beseen in Figure 53. Note that flow and heat flows here are calculated not in TPP units but so as we have programmed in the condenser model, I.e., flow – in t/h, Q – in MW, Qf – in kW/m2.

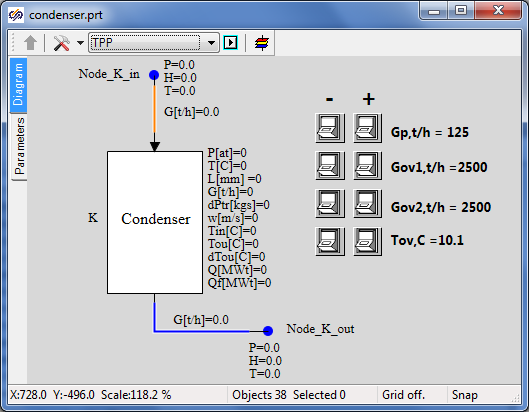


Figure 53. Display of model parameters in diagram window

### Properties of boundary conditions

1. Set properties in **“NodeG\_K\_in”** boundary condition:

Flow: **“Gp/3.6”**

Enthalpy: **“495»”**

Initial enthalpy: **“Self.H”**

Flow area, hydraulic diameter, wall thickness, section length: **“1”** (these properties are not important at this stage, water supply with preset flow and enthalpy is required from the boundary condition).

1. Set properties in **“NodeG\_K\_out”** boundary condition:

Flow: **“-Gp/3.6”** (minus is important since flow here is directed to the boundary condition)

Enthalpy: **“32”**

Flow area, hydraulic diameter, wall thickness, section length: **“1”**

Initial enthalpy: **“Self.H”**.

Elevation: **“-15”**.

1. Boundary nodes G of heat exchangers:

Flow: **“-Gov1/3.6»**”and “**-Gov2/3.”**, correspondingly.

Enthalpy: **“Tov”**

Initial enthalpy: **“Self.H”**

1. Boundary nodes P of heat exchangers:

Pressure: **«1»**

Enthalpy: **“Tov”**

### Properties of condenser lines

There are totally 4 common-mode channels – steam supply to condenser, condensate drain and by one channel per each heat exchanger. We have set parameters for heat exchanger channels (inside the condenser model) by program method. Now we will set properties in channels outside the condenser:

|  |  |
| --- | --- |
| Channel **“Ch\_K\_in”** | Hydraulic diameter: **«2.0»**  Flow area: **«3.1415»**  Direct local resistance: **«0.1»**  Reverse local resistance: **«0.1»**  Heat transfer surface: **«31.4159»**  Length. **«5.0»**  Wall Thickness: **«0.01»** |
| Channel **“Ch\_K\_out”** | Hydraulic diameter: **«1.0»**  Flow area: **«1.0»**  Direct local resistance: **«1.0»**  Reverse local resistance: **«1.0»** |

### Properties of 3V TPP tank and the project as a whole

1. Go inside the condenser submodel and in the properties of 3V tank make sure that the properties set are the same as depicted in Figure 54. Change the properties:

Number of vertical tubes: **“n”**.

Fix the index: **“Yes”**.

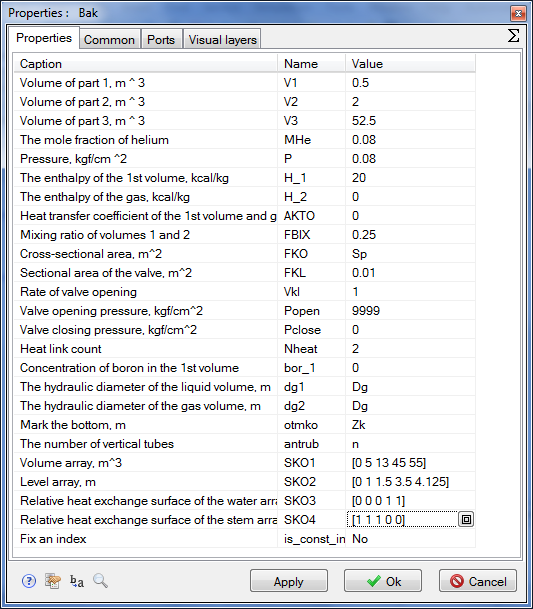


Figure 53. Properties of condenser submodel 3V tank

1. Now we need to set properties in points – holes in the tank. Go to the upper hole properties (steam supply to condenser) and set new properties for the following parameters:

|  |  |
| --- | --- |
| Condenser upper hole (steam supply) | Initial pressure: **“0.036”**.  Initial enthalpy: **“532.28”**.  Hydraulic diameter: **“1”**.  Flow area: **“1”**.  Length of node: **“1”**.  Heat transfer surface: **“3.728”**.  Elevation: **“Zk+Hv+Hp”**.  Material: **“Ст20”**.  No. of volume: **“Steam”**. |
| Condenser lower hole (condensate drain) | Initial pressure: **“0.036”**.  Initial enthalpy: **“27.7”**.  Hydraulic diameter: **“2.0”**.  Flow area: **“18.84”**.  Length of node: **“0.1”**.  Wall thickness: **“0.02”**.  Heat transfer surface: **“0.628”**.  Elevation: **“Zk”**.  Material: **“Ст20”**.  No. of volume: **“Lower water”.** |

1. In the next step go to **“Calculation parameters”** dialog window and change three properties there:

TPP project name: **“kp\_3200”**.

Integration step for energy equations: **“0.125/4”**.

Integration step for motion equations: **“0.04/16”**.

For checking and comparison see Figure 54.

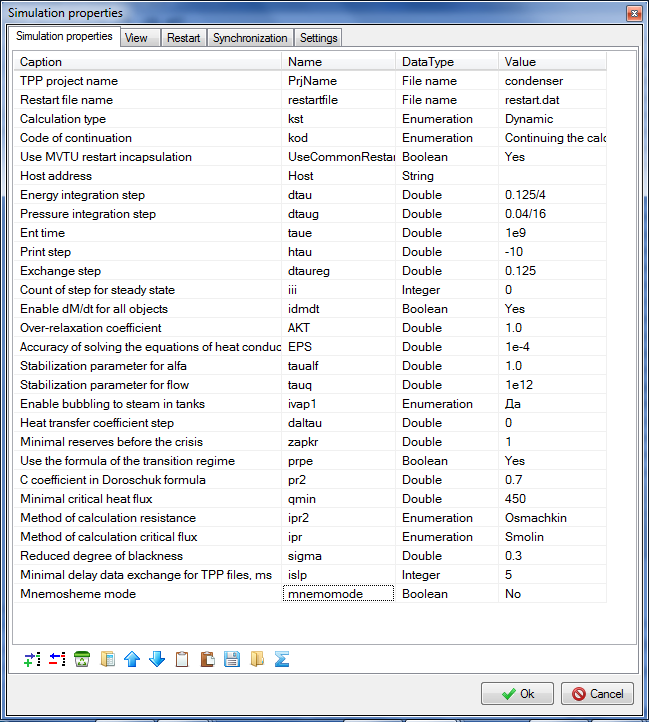


Figure 54. Condenser submodel calculation parameters

### Rated state

Now, in case all has been correctly executed, after starting the model for calculation, stationary state of the model shall be set within 150-300 seconds, the same as shown in Figure 55. In order to debug and check the stability of the model you can use buttons, build up additional graphs for parameters interesting for us and change boundary conditions in the process of calculation, meanwhile watching changes in the state of the condenser.

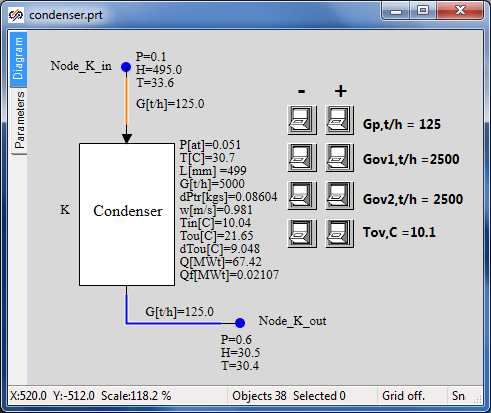


Figure 55. Stationary state of condenser model

So, we have created the model of turbine condenser and placed it inside the submodel and achieved the stable state corresponding to nominal parameters of the condenser (in terms of pressure in the condenser, condensate temperature and hydraulic resistance). Then we will have to create models for heaters PN-100, PV-280-1 and PV-280 that are to a large extent similar both to each other and to the model of the condenser.