# Integration of Models, Generation of Integrated STP Model

## Main Steam Model Generation

### Model Description

In this tutorial example of generation of integrated STP model (process of integration of many models into one) we will start with the main steam system. We will generate a main steam model consisting of two pre-generated models, i.e., the flowing part and main condenser KP-3200, on a separate TPP page.

If we recall pre-generated models or open those in SimInTech it can be seen that there is a boundary node of P type with 0.05 kgf/cm2 pressure in the flowing part at the right, and steam is flowing to that with 125 t/h flow and parameters close to nominal ones. In the condenser model at the top there is a boundary node of G type with 0.1 kgf/cm2 pressure, and steam is delivered from that with the same flow of 125 t/h and with about the same nominal parameters. We will have to combine these diagrams into one and achieve their interoperation in nominal mode.

When integrating the models into one, parameters at boundary nodes shall be theoretically the same but practically parameters can be slightly different from each other due to errors of numerical simulation or round-off of set properties by other diagram elements, or due to input errors of one or other properties (for example, our pressure is different by 0.05 kgf/cm2). Therefore, when combining models, one shall ever monitor a new state and practically every time debug a newly obtained diagram. Nevertheless, since the models have been individually debugged, debugging of a new diagram is reduced to minimum manipulations and changes of parameters. Exactly due to this fact we have first created all (or about all) diagram elements one by one and now are starting to integrate those. If all initial data and all parameters had been known by us in good time a new thermohydraiulics diagram of STP could have been immediately created. No one acts in such a manner since all initial parameters are seldom available, and the most essential thing is that the diagram cannot be completely comprehended in the beginning of design.

So, we have generated a sum of two models in one file, i.e., the flowing part ad condenser.

Important note: continuation of the story will be told considering the fact that the SimInTech user (reader) is more or less skillful, and we will not have to dwell on elementary technical details, such as what button shall be pressed and in what place one parameter or other can be changed. Only important benchmarks, we will have to pass through for generating a working model, will be pinpointed.

### STP Model File, Version 01

Open the file with flowing part model created earlier and save it into **“**C:\KTZ\Turbine\ТK‑35.prt” file.

Rename the project descriptive parameters: change TPP project name in the calculation parameters for: **“tk\_35”**. Steps of integration shall be set in **“0.125/4”** for energy equations and in **“0.125/16”** for motion equations.

Place a “TPP submodel” new element on the diagram and transfer the whole flowing part model to inside of the page. Place one more TPP submodel nearby and transfer the control of boundary conditions to inside that. Change properties of the pages to make their appearance look like Figure 88. Assign page names as **“MSS”** and **“BCC”**.

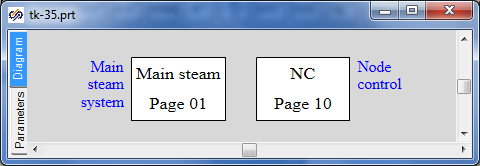


Figure 88. Main Steam System, Beginning of STP Integrated Model Generation

Save the project (again).

Therefore, we have just generated a new file with a copy of the flowing part model divided into the model itself and the basic manual control of boundary conditions. Further then we will attach other models and place them on correspondent submodel pages, while concentrating the control on the **“BCC”** (boundary conditions control) page.

### Global Parameters

3 (three) global parameters are used in the flowing part model (“Pstg”, “Gstg” and “Tstg” – parameters of steam upstream of the turbine); all those have been already transferred to the new main steam model.

4 (four) global parameters are used in the condenser model; note that we do not need one of these parameters (“Gp” – steam flow to condenser) since the flowing part produces a well defined steam flow – the same 125 t/h when in nominal mode.

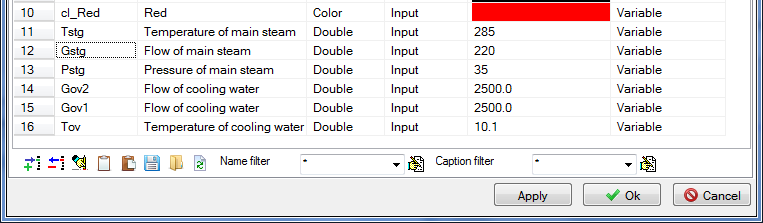


Figure 89. Main Steam System, Six Global Signals

Thus, three more global parameters shall be added to the diagram (“Tov”, “Gov1”, “Gov2”); those can be copied from the condenser model generated earlier. Compare the obtained list of signals with Figure 89.

### Integration of the Flow Part and Condenser Models, Structure

Now copy the WHOLE condenser model onto the “MSS” page as depicted in Figure 90.

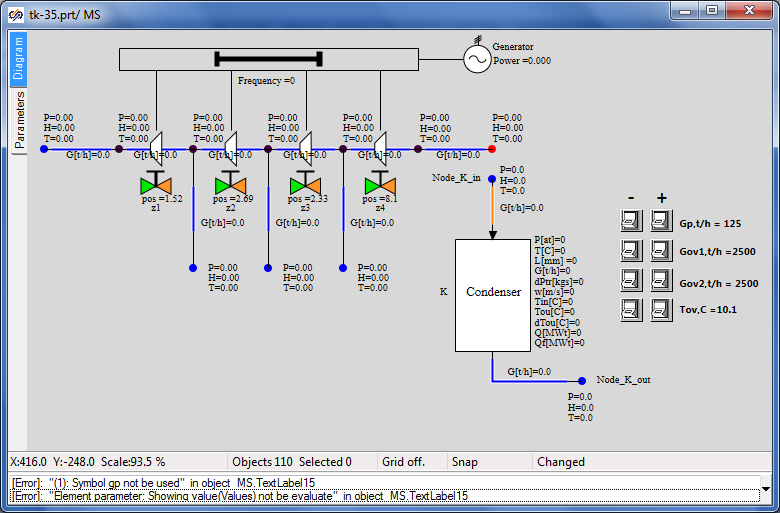


Figure 90. Integration of the Flow Part and Condenser Models

Then SimInTech generates an error due to no **“Gp”** global signal in the project: such a variable is not available and it can be used neither in “TextLabel” element next to the button, nor as the value of G boundary node parameter.

Since this signal is not required anymore, delete related buttons (**“Gp”** zoom-in and **“Gp”** zoom-out) and their caption, delete boundary node “NodeG\_K\_in” and also delete boundary node P of the flowing part – we do not need all these elements anymore.

The connection could have been implemented in two ways – a new internal node could have been set instead of two boundary nodes and channels could have been connected from the flowing part and from the new node to the condenser. Nevertheless, since the new internal node is not required here (the flowing part has been so modeled as if the boundary node P is the condenser, and set relevant parameters there; while the condenser has been modeled so that the boundary node G was the STP outlet) we will delete one of the channels and connect the condenser directly to the marginal internal node of the flowing part.

Delete the marginal channel of the flowing part and connect the condenser inlet channel “Ch\_K\_in” to the marginal internal node of the flowing part. Transfer the boundary conditions control buttons onto the “BCC” page. Compare the result with Figure 91.

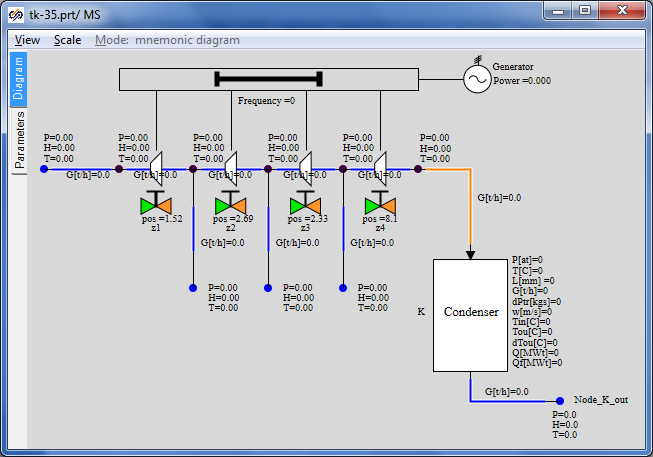


Figure 91. Structure of Main Steam System (Flowing Part + Condenser)

Transfer the script text for button controls used to set cooling water parameters from the conden­ser model. Insert the text into the “Parameters” tab on the “BCC” page, add necessary corrections for buttons names, delete two rows related to parameter “Gp”, see Figure 92.

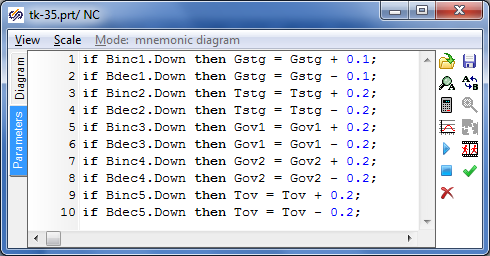


Figure 92. Button Control of Main Steam System Boundary Conditions

### Display of Parameters in Diagram Window

All relevant parameters have been already displayed in the diagram window.

### Properties of Main Steam Model Elements

If we try to launch the diagram for calculation in its current variant it will not be launched – we “have forgotten” that signal **“Gp”** is used in the condenser output node G as well, which will be reminded by SimInTech in case of an attempt to launch the diagram for calculation. To ensure correct operation in the nominal mode we will set the condensate outlet flow from the condenser equal to the steam inlet flow, i.e., equal to the flow in channel **“Ch\_K\_in”**. Change this parameter; note the “minus” sign:

|  |  |
| --- | --- |
| Boundary node of “G” type at the condenser outlet | Flow: **“--Ch\_K\_in.g”** |

The other parameters have been already set and have been set correctly, nothing is to be changed.

### Nominal State of Main Steam System

Now launch the diagram for calculation, nominal state shall be obtained with parameters as depicted in Figure 93. Parameters obtained in this state are similar those ones that have been obtained one by one in the nominal mode of operation of the flowing part and condenser submodels.

Let us remind that that the flow for all boundary conditions is still “squeezed” in the model, while pressure in flowing part nodes is defined via a degree of opening/closing of gates z1-z4.

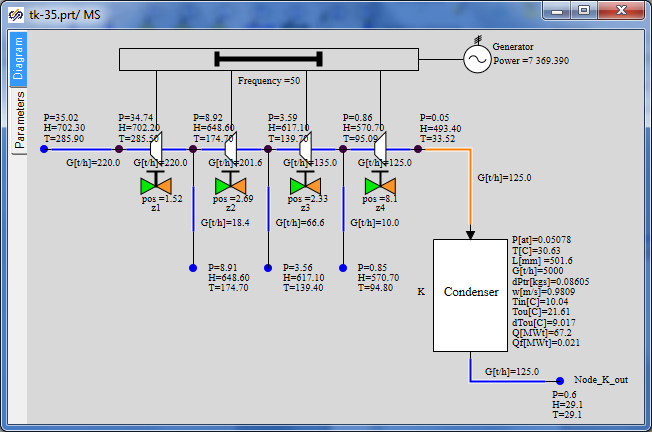


Figure 93. Nominal State of Main Steam System

### Control of Generator and Turbine Rotor Parameters

Let us introduce two sensors on the diagram for further use – turbine rotor rotation speed sensor and generator power sensor. Let us set sensor names as “SE01G11B1” and “SE01N01B1”, correspondingly. Units: “Hz” and “MW”.

Place two elements of “TPP node pressure sensor” type on the rotor and on the generator and change their properties (control point properties) according to Figure 94 and Figure 95. Set parameters according to the Table:

|  |  |
| --- | --- |
| Rotor rotation speed control point | Object name: “F\_K5”  Element type: **“TPP rotor rotation speed sensor”** |
| Generator power control point | Object name: “Nael”  Element type: **“Power sensor in TPP”** |

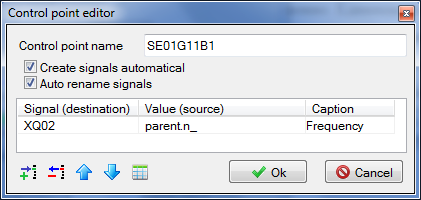


Figure 94. Rotor Rotation Speed Control Point

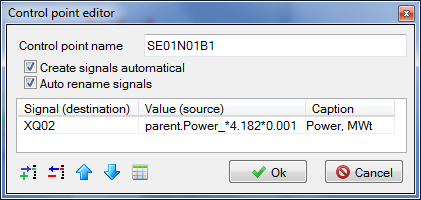


Figure 95. Generator Power Control Point

We have set the “create signals automatically” properties. It means that two new signals with unique names of “SE01G11B1\_XQ02” and “SE01N01B1\_XQ02” type have been created in the database. You can enter the database and make sure of that.

Now let us display the power parameter in the display window to continuously see the power value in the process of model calculation. To that end, enter the main menu item **“Calculation” → “Data manager”** and create the “Calculation results” category there with one display window, wherein set one parameter “SE01N01B1\_XQ02”, see Figure 96.

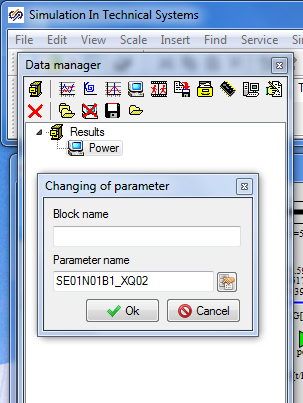


Figure 96. Generation of Parameter Display New Window

Now click the “Power” element in the data manager to see a new display window, in which electric generator power value will be displayed in the process of calculation in MW, see Figure 97.

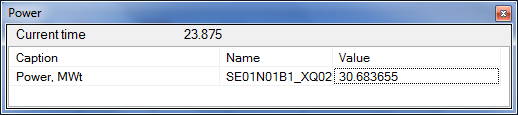


Figure 97. “Generator Power” Parameter Display Window

Save the project under the name of **“C:\KTZ\Turbine\ТK-35-version-01.prt”**. Immediately resave it under the other name of **“C:\KTZ\Turbine\ТK-35-version-02.prt”**. We will not change the first version (**“ТK-35-version-01.prt”** file), while the integration process will be conducted in the second one. This division into versions is convenient when return to an intermediate point is required during generation of a large model.

## Connection of the Main Steam System and Condensate Equipment Group

### Model Description

The next step in the process of integration of submodels will include connection of condensate group of equipment – PLP-1 and condensate pumps.

Connection points will be as follows (totally 3 (three) ones):

* condensate pumps will supply water to the heated water inlet to PLP-1;
* condensate drained out of the condenser will be supplied to the suction of the condensate pump groups;
* we will connect boundary condition G of the third extraction with the heating steam inlet in PLP-1 in the flowing part submodel (main steam system).

Condensate equipment group (pumps and PLP-1) will be placed on a separate TPP page, i.e. in a separate TPP submodel. We will arrange connections between TPP pages using “To TPP memory” and “From TPP memory” elements.

### STP Model File, Version 02

New TPP project has been already created in the previous subsection. Open that: **“C:\KTZ\Turbine\ТK-35-version-02.prt**” file.

Place a new “TPP submodel” element on the diagram (to speed up the process it will be easier to copy one of existing pages and delete its internals). Change properties of the new page to make their appearance look like Figure 98. Set **“CEG”** row as the new page name.

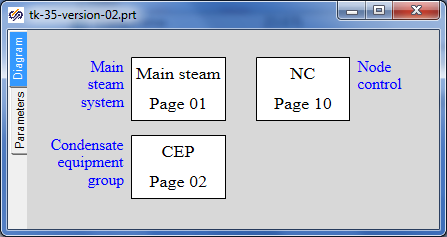


Figure 98. New TPP Page for Condensate Equipment Group

Save the project. We have just created (re-written) a new file with a copy of the main steam system and with a blank for condensate equipment group.

### Global Parameters

To ensure correct operation of PLP-1 model two global signals are required: **“Gplp**” and **“Tplp”**. Add those to the project by copying them from PLP-1 model, see Figure 99.

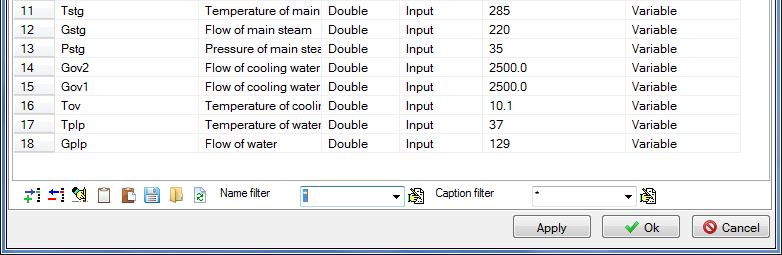


Figure 99. Additional of Gplp and Tplp signals

### Structure of Connection of Condensate Equipment Group

Now copy the PLP-1 condenser model and model of condensate pumps onto the **“CEG”** (condensate equipment group) page as depicted in Figure 100.

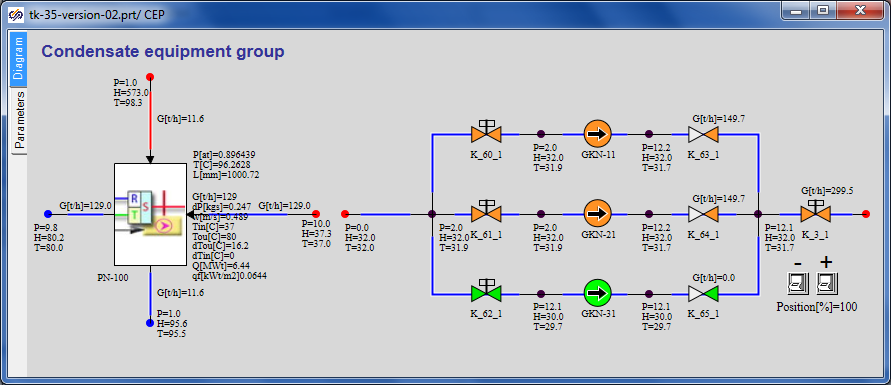


Figure 100. Generation of Condensate Equipment Group

At the same time do not forget the button script – copy that too. Transfer the control buttons for PLP-1 global signals to the “BCC” page. Keep the gate “K\_3\_1” control buttons on the same page (equivalent of local control).

If the diagram is launched for calculation now then due to no links among PLP-1, pumps and main steam system all these parts will be calculated independently and remain in the same nominal state, in which those have been kept during creation of each submodel.

Let us start connection with the very simple thing – i.e., connect the condensate pumps group with the heater. To that end we will need to remove both boundary conditions (at the right from the heater and at the left from the pumps), and also to remove the common mode channel supplying water to PLP-1. Instead of these three elements add the TPP internal node to the diagram (copy the node up to gate **“K\_3\_1”**) and integrate the two models into one. The example can be seen in Figure 101.

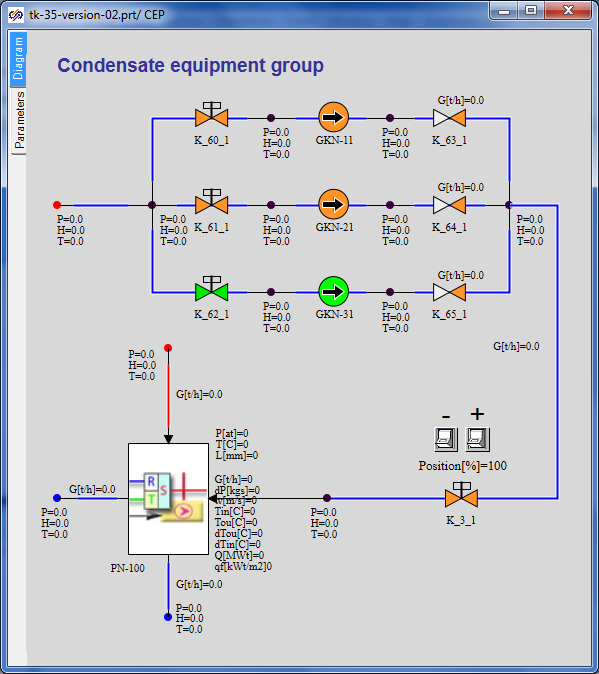


Figure 101. Connection of PLP-1 and Condensate Pumps

Elevation of all internal nodes downstream of the condensate pumps shall be set to zero.

After this connection, about nothing has been changed on the diagram – heated water flow is “squeezed” by the boundary condition downstream of PLP-1 and cannot be dynamically changed, i.e., head characteristics of the pumps do not yet fairly “work”. Those, of cause, participate in the calculation but cannot provide a total flow different from the one set in the boundary condition.

Condensate drain and its supply to the suction of condensate pump group will be the next connection point. Here we will use **“To TPP memory”** and **“From TPP memory”** blocks to connect elements on different pages of the TPP model.

Go to the **“MSS”** page and replace the **“NodeG\_K\_out”** boundary condition at the condenser outlet with **“To TPP memory”** element. Rename this variable as **“To GKN suction line”**, see Figure 102.

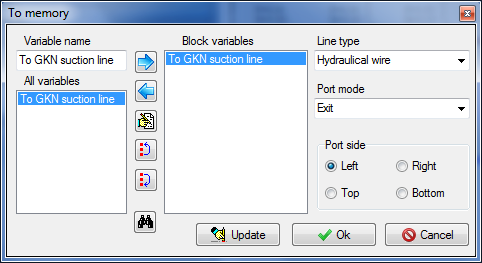


Figure 102. Creation of a New Variable in TPP Memory

Note that an asterisk symbol precedes the name of this variable on the diagram – it means that the variable is being created but is not yet used anywhere on the diagram, and the diagram cannot be launched for calculation while in such state since there is no “counterpart” to this block in another place of the diagram.

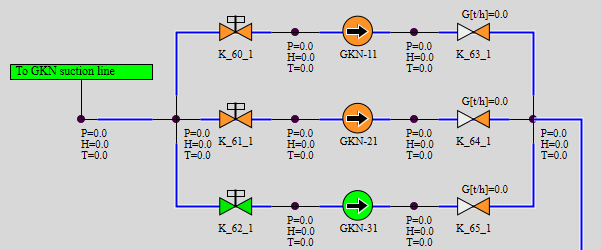


Figure 103. Connection of Suction Line of Condensate Pumps with Condenser

Go to the **“CEG”** page: place an internal TPP node instead of boundary node P at the suction line of the pumps (the internal node next to the boundary condition to be deleted can be copied) and **“From TPP memory”** element, on setting the same **“To GKN suction line”** name. After that the asterisk will disappear before the name of **“To TPP memory”** element on the main steam system page.   
It should be noted that “from memory” and “to memory” blocks do not generate any TPP objects; those just interconnect those via the mechanism of creation of a new variable in the program memory. I.e., these elements are free of any internal dynamical state variables.

The third connection point – extraction to PLP-1 – shall be also arranged using the “to memory”/”from memory” mechanism. Delete the boundary conditions on the line of flowing part third extraction (boundary node G) and the boundary condition on steam supply to PLP-1 (boundary node P).

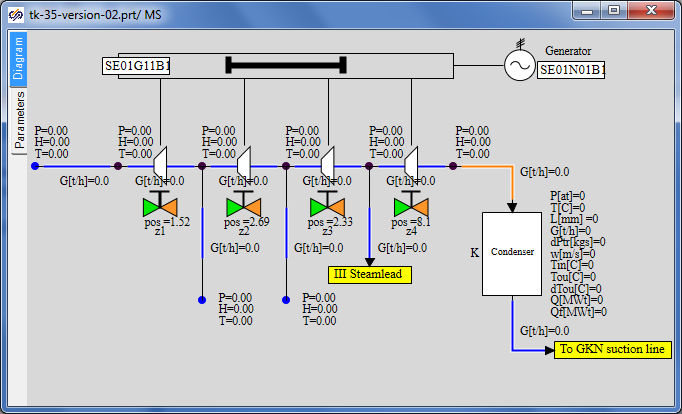


Figure 104. Connection of the Third Extraction and PLP-1

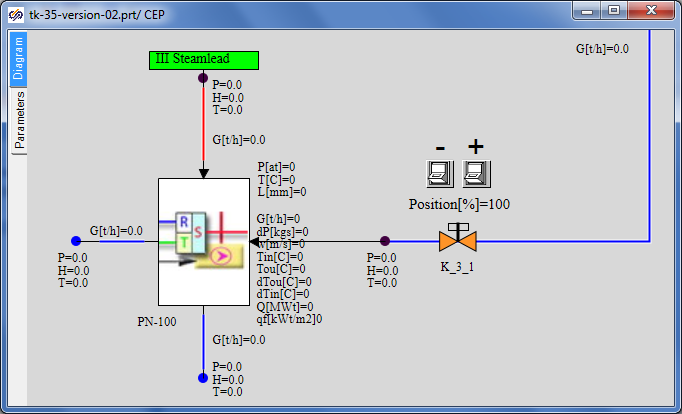


Figure 105. Connection of PLP-1 and the Third Extraction

Replace node G with **“To TPP memory”** element named as **“Extraction III”** and replace node P with internal node TPP (copy the node between LP heater and gate **“K\_3\_1”**) and with **“From TPP memory”** element named as **“Extraction III”**. For examples see Figure 104 and Figure 105.

### Display of Parameters in Diagram Window

All relevant parameters have been already displayed in the diagram window.

### Properties of Model Elements

Parameters of model elements have been set correctly, nothing is to be changed. On launching the diagram for calculation, a state close to nominal shall be set.

### System Nominal State

In this step of integration it is important for us to obtain stationary and stable state of the system. If everything is done correctly then the final state will be close to the nominal state since many parameters are squeezed by boundary conditions and do not allow the model to strongly deviate from the nominal. Actual debugging of the model, obtaining of the stationary state, error detection and model completion will be possible in further steps of integration when the model becomes more complicated and “flexible”, level regulators and other elements are added.

Compare the state of the system with Figure 106.

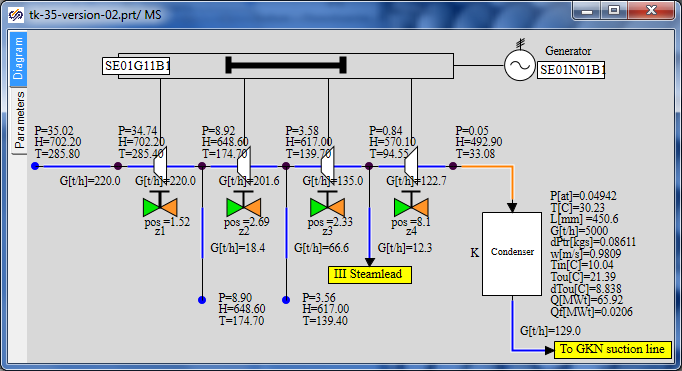


Figure 106. Stationary Calculation of Connection of PLP-1 and the Third Extraction

## Connection of Feedwater System

### Model Description

Connection of the feedwater system – PHP-2 and PHP-3, as well as feed pumps group, is the next step in the process of integration of submodels.

Connection points will be as follows (totally 4 (four) ones):

* heated water from PHP-2 will be supplied for further heating to PHP-3;
* feed pumps will supply water to the heated water inlet in PHP-2;
* we will connect the second extraction with the heating steam inlet in PHP-2 in the flowing part submodel (main steam system);
* we will connect the first extraction with the heating steam inlet in PHP-3 in the flowing part submodel (main steam system).

The feedwater system (pumps, PHP-2 and PHP-3) shall be placed in a separate TPP model. Connections between TPP pages will be arranged using **“To TPP memory”** and **“From TPP memory”** elements.

### STP Model File, Version 03

Open the project, version 02, (“**C:\KTZ\Turbine\ТK-35-version-02.prt**”) and save it in a new file named as **“C:\KTZ\Turbine\ТK-35-version-03.prt**”.

Arrange a new **“TPP submodel”** element on the diagram. Change properties of the new page to make their appearance look like Figure 107. Set **“FWS”** row as the new page name. Page name: **“Feedwater system”.**

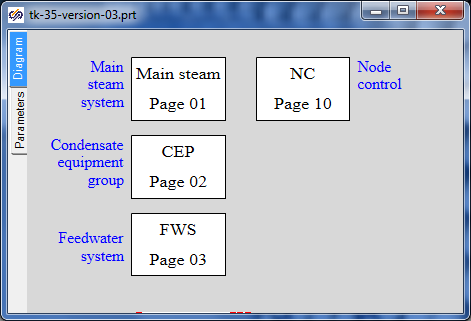


Figure 107. Feedwater System Submodel

### Global Parameters

To ensure operability of our PHP-2 and PHP-3 model 6 global signals, generally, are required – water flow and temperature for every heater, as well as steam pressure in correspondent extractions. Nevertheless, since we are interconnecting the heaters via water, water parameters for PHP-2 will be calculated as input ones in PHP-3. And steam pressure will be directly taken from extractions. Therefore, we need to add only two global signals: **“Gphp3**” and **“Tphp3”**. Add those to the project by copying them from PHP-3 model and renaming, see Figure 108.

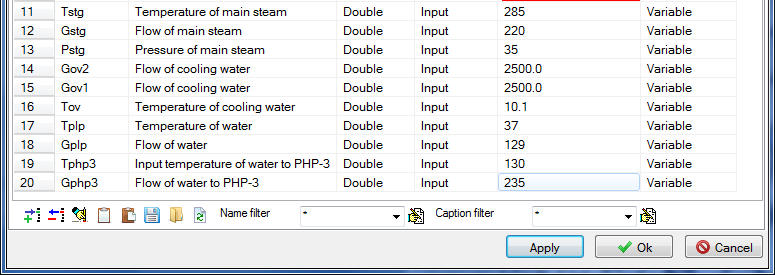


Figure 108. Addition of Signals Gphp3 and Tphp3

### Structure of Connection of Feedwater System

Now copy PHP-3 (PV-280) submodel onto the **“FWS”** page from correspondent projects, at the right of that copy PHP-2 (PV-280-1) submodel and above that – the model of feed pumps, see Figure 109.

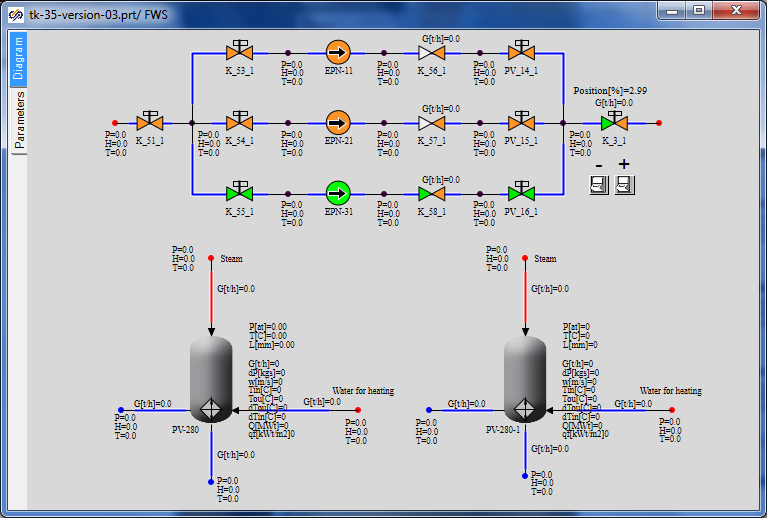


Figure 109. Generation of Feedwater System Submodel

At the same time do not forget the button script – copy that too.

Then transfer the control buttons for PLP-3 global signals to the “BCC” page (along with related text; delete the steam pressure control button). Keep the gate “K\_3\_1” control buttons on the same page (equivalent of local control). Delete the control buttons for PHP-2 boundary conditions (delete also the text – it is not required anymore).

Let us start the connection with an evident thing – let us interconnect the heaters via the heated water path. To that end delete the both boundary conditions (at the right from PV-280 and at the left from PV-280-1) and place the TPP internal node there (copy the node up to gate **“K\_3\_1”** and set the elevation of a new internal node to zero), then interconnect the both models of heaters. The example can be seen in Figure 110.

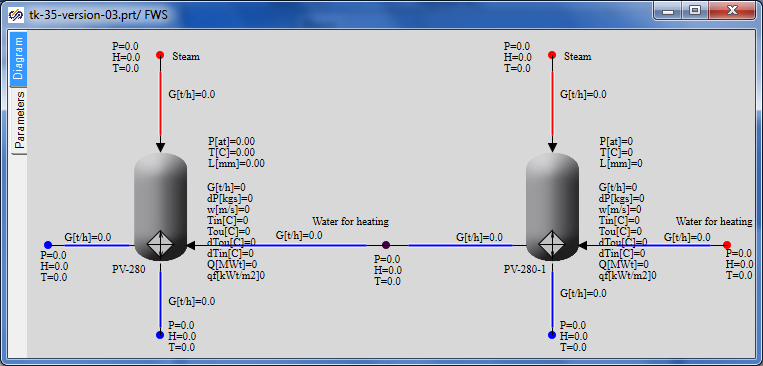


Figure 110. Connection of PHP-2 and PHP-3 via Water

Then connect the group of feed pumps and PV-280-1 inlet (PHP-2). To that end delete the boundary conditions from the both sides in the same manner, add a new internal node (copy the node close to gate **“K\_3\_1”**) and interconnect the submodels by means of hydraulic links. Everything here is the same as connection between PHP-3 and PHP-2.

The next connection points are steam extractions (first and second). Go to the main steam system page and add two new “To TPP memory” elements with port names “Extraction I” and “Extraction II” there. Use the first element to replace boundary node G and do not yet delete the second boundary node G – just disconnect it from the steam extraction channel.

The situation here is as follows: we need to remove not all steam of the second extraction, which is 66.6 t/hour in the nominal mode, in PHP-2 but only a part of that: 13 t/hour. Thus, add one more internal node in this place and a branch using two common mode channels; an example of arrangement of elements can be found in Figure 111. Properties of the node and channels shall be selected considering the following: we will copy the node from the previous link (between the pumps and PHP-2), while the second extraction channel shall be copied for properties of the channels.

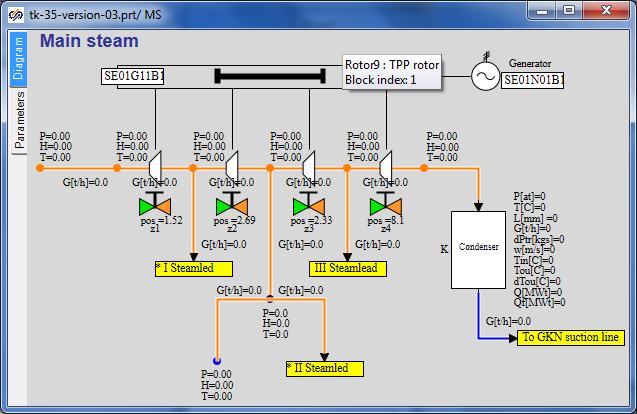


Figure 111. Arrangement of Steam Extraction I and II

On doing that, delete boundary conditions for steam supply from extractions to heaters on the feedwater system page and place **“From TPP memory”** elements with relevant variable names there. Make sure that asterisks have disappeared on the main steam system page.

Now connections of submodels in this subsection are over, we can try to launch the diagram for calculation. Meanwhile, SimInTech will inform us on some errors that shall be rectified: we have changed the name of global parameters for PHP-3, while old names have remained in boundary conditions. Replace those with new ones: **“Gphp3”** and **“Tphp3”** (instead of **“Gv”** and **“Tv”**). Set pressure in tanks and internal nodes equal to **“0.96”** (instead of **“Pp”**).

### Display of Parameters in Diagram Window

All relevant parameters have been already displayed in the diagram window.

### Properties of Model Elements

Parameters of about all elements have been set correctly (parameters of new elements were set as new elements were added) but some changes and tests shall be done. When copying diagrams from separate submodels into a common one, those diagram elements, whose names are the same as the names of existing elements, were renamed.

In our case PHP-2 and PHP-3 models had the same internal names of channels and points, which were renamed. Due to the above all boundary nodes G shall be checked for what is set as flow in those, and all shall be corrected for true values. Do it.

Also correct elevations of points downstream of all pumps: points shall be sunk upstream of the pumps and be at elevation 0 downstream of the pumps.

Set the flow equal to **“-(66.6-13**)/3.6” in boundary node G at the second extraction since we have to extract 13 t/h for the second heater, while 66.6 t/h corresponds to the nominal flow rate at the second extraction.

Water flow supplied for heating to PLP-3 and actually extracted from the condenser shall be changed from 129 for 125 t/h (in global signals).

### System Nominal State

After launching of the system for calculation, strong fluctuations are possible in the initial period of time, since now we have connected many points together and the system is “seeking” for a new stationary state.

In 100–300 seconds the stationary state shall be obtained, which can be different from the nominal one. It depends on particular settings of all diagram elements – i.e., channels, points, pumps, etc. For example, the following state was obtained in our variant immediately after connection (see Figure 112).

In that extraction flows do not match the nominal state, although are close to that in terms of quality; but the most important thing is that the total amount of all extractions and condensate extraction from the condenser exceeds the main steam supply flow that causes a continuous drop of the condenser level. It has happened because we have “forgotten” to change the flow in boundary node G of the second extraction and extractions of other boundary nodes G on heaters.

Thus, in any case, it is required to conduct an additional test and debugging of the diagram in order to obtain the nominal state.

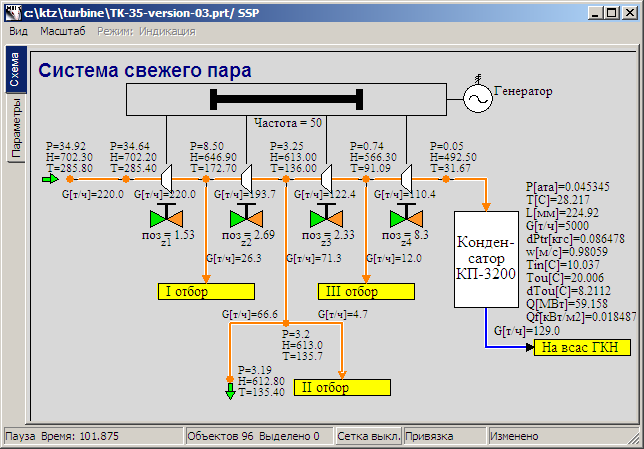


Figure 112. State of Main Steam System after Adding Steam Extractions

Tank levels are the most inertial (integral, as it can be said) values in this version – those require maximum time to reach the nominal value. So, after rectifying all gross errors and after 2000 seconds of calculation the following state of the system was obtained (see Figure 113); besides, levels in all tanks became close to 1 m and practically were not changed, i.e., the system had reached its stationary state, from which it can be finely readjusted and returned to the nominal state.

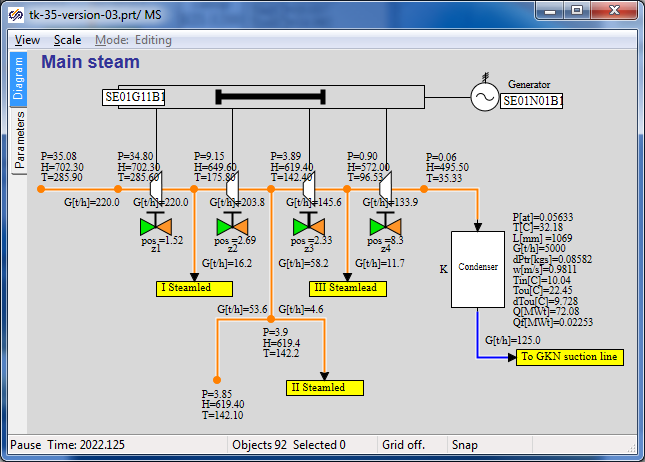


Figure 113. State of Main Steam System after Rectification of Gross Errors

Now adjustment of the system can be completed using various techniques: change of positions of gates, resistance in one or other channels, etc. We will not proceed like this. First: to conduct such “fine” adjustment of the system we have not enough initial data. Second: in this step of integration too many parameters are “Squeezed” by boundary conditions and in the next steps we will in any way have to readjust the system for new stationarity. I.e., fine readjustment will be conducted later after deaerator is added and some boundary conditions of G type are removed.

Now try to operate independently with the diagram and reach the stationary state closer to the nominal state than the one depicted in Figure 113.

For comparison refer to another screenshot with different position of gates z1,z2,z3,z4 – see Figure 114. Here PLP-1 nominal flow (10 t/h) has been successfully set.

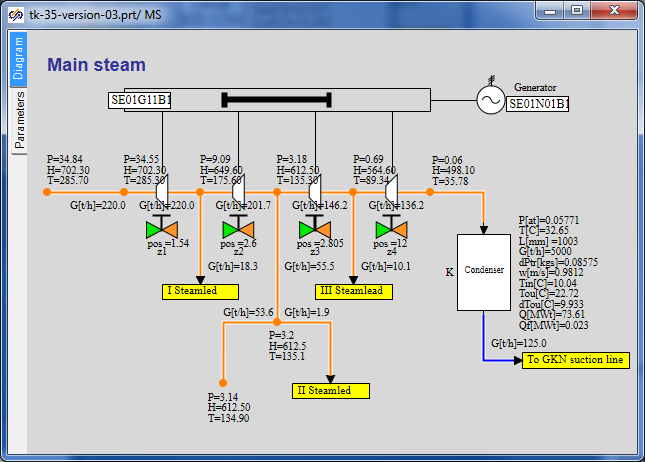


Figure 114. State of Main Steam System after Change of Positions of Gates

## Connection of Deaerator

### Model Description

Now we will add the deaerator model step by step to our STP model.

Connection points to the deaerator will be as follows (totally 3 (three) ones at this step):

* deaerator extraction will be added to the second steam extraction beside PHP-2 – the steam line will connect the steam extraction with the deaerator top volume;
* drained condensate from the deaerator will be supplied to the suction line of the group of feed pumps (and then to heating in PHP-2 and PHP-3);
* water heated in PLP-1 will be supplied to the deaerator top volume.

We will place the deaerator model in a separate TPP model. Connections between TPP pages will be arranged using **“To TPP memory”** and **“From TPP memory”** elements.

### STP Model File, Version 04

Open the project, version 03, (“**C:\KTZ\Turbine\ТK-35-version-03.prt**”) and save it in a new file named as **“C:\KTZ\Turbine\ТK-35-version-04.prt**”.

Arrange a new **“TPP submodel”** element on the diagram. Change properties of the new page to make their appearance look like Figure 115. Set **“DEAER”** row as the new page name.

The whole submodel can be just copied from the deaerator project into the integrated STP model file.

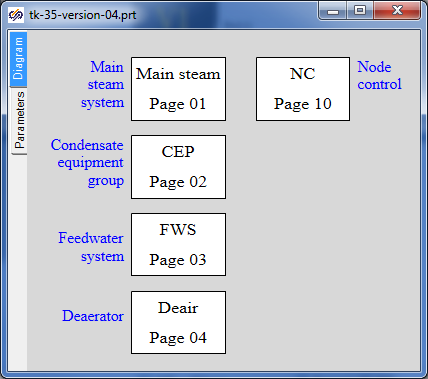


Figure 115. Addition of Deaerator Submodel Page

### Global Parameters

To ensure operability of the deaerator model no global signals are required since we have generated the model without those. Thus, nothing is to be changed or added in the global parameters.

### Structure of Connection of Deaerator Submodel

Now copy the deaerator submodel from a corresponding project onto the **“Deaer”** page and then we will start connecting the deaerator model with the STP model.

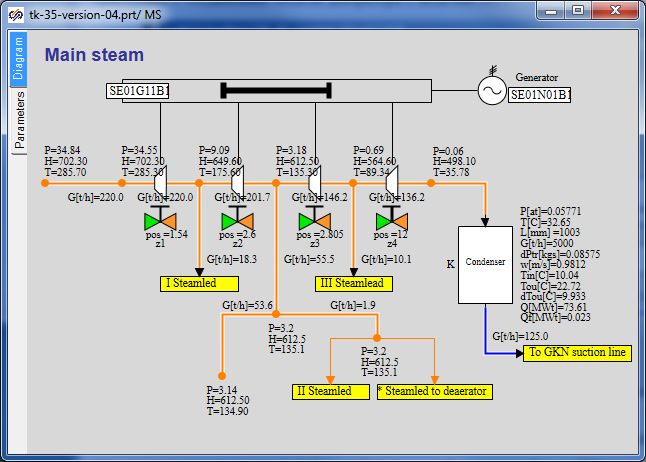


Figure 116. Addition of New Extraction to Deaerator

Steam for the deaerator will be extracted in the same point the steam for PHP-2 has been extracted.   
To that end a new TPP internal node (copy a neighboring node, the new node will have the same parameters) and new **“To TPP memory”** block named as **“Extraction to Deaer”** shall be added in the point of extraction. Interconnect the both **“To TPP memory”** blocks with the new internal node, see Figure 116.

Place the **“From TPP memory”** block on the deaerator submodel in place of boundary node P. Set the same name for that (**“Extraction to Dear”**). Make sure that the asterisk has disappeared in the twin **“To TPP memory”** unit.

Now arrange connection of drained condensate from deaerator to the suction line of feed pumps in the same manner. Names of **“To TPP memory”** and **“From TPP memory”** blocks shall be set as **“To EPN suction line”**. New point (internal TPP node) instead of boundary node P on the submodel of feed pumps should not be set (as well as on the deaerator model) – tank hole is directly connected to the channel on the diagram of feed pumps (channel with gate **“K\_51\_1”**).

Go to the submodel of condensate equipment group and create a link between water heated in PLP-1 and the deaerator. Name of the variable in the TPP memory shall be set as **“Condensate to Dear”**. Since there is an internal node (tank hole) on the deaerator tank here, a new internal node is not to be added. On the contrary, the channel on the deaerator submodel can be deleted, then heated water will be directly supplied to the deaerator tank via the mechanism of a variable in the TPP memory.

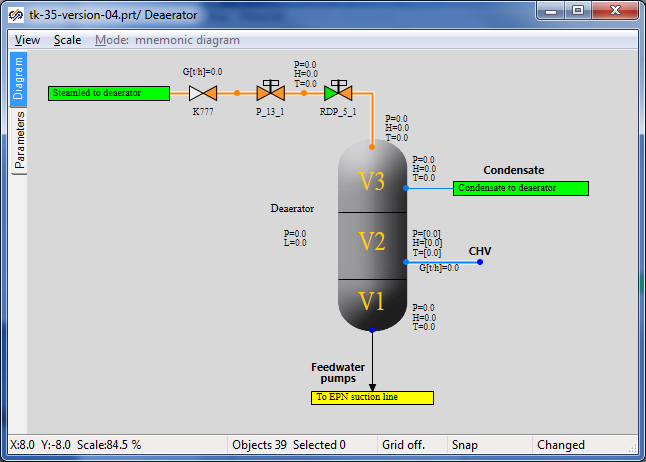


Figure 117. Deaerator Submodel with Links Connected

Approximate appearance of the deaerator submodel with three new connections can be found in Figure 117.

Now we will have to do accurate debugging of the model since the deaerator submodel has been created without setting any stationary and nominal state. We will have to revise the whole diagram and, when required, readjust some elements (element properties) to obtain stable nominal state.

### Display of Parameters in Diagram Window

All relevant parameters have been already displayed in the diagram window.

### Properties of Model Elements

Parameters of all model elements have been correctly set (parameters of new elements were set as new elements were added). Now we are not going to change particular things but in the process of debugging of the nominal state required changes will be introduced into the model.

### Stationary State of the Model with Connected Deaerator Model

Let us launch the diagram for calculation, wait for stationarity is obtained and see what its difference from the nominal state is. The next discourse is conducted according to a model created from zero in parallel with the writing of this training manual. In your case another stationary state can be obtained. Corrections we have to do in the process of development of the training manual are described below:

Let us start with the flowing part. Set the extraction value equal to **“0”** in boundary node G (in the second extraction) (in the next section we will add network water heaters here; now the whole extraction will be supplied to the deaerator).

Set the initial position of control gate **“K\_3\_1”** equal to **“6.5%”** in the model of condensate pumps.

Replace valve **“K\_51\_1”** with a check valve in the model of feed pumps – a check valve is to be fitted here – an error was done during development of the submodel. Set the initial position of gate **“K\_3\_1”** equal to **“25%”** in the same model. All points of the submodel of feed pumps shall be at elevation **“0”**.

Set volume values in the deaerator tank model: **“5”, “40”, “200”** m3 and section area equal to 7 m2. Valve **“RDP\_5\_1”** shall be in position “100%”.

Beside these corrections some “cosmetic” improvements have been implemented. The manual cannot describe all up to the last point.

As a result after 1000 seconds of calculation the following stationary state was obtained – see Figure 118. It can be seen from the Figure that flows in extractions are different from nominal ones, steam parameters in extraction points with some error are similar to the nominal values.

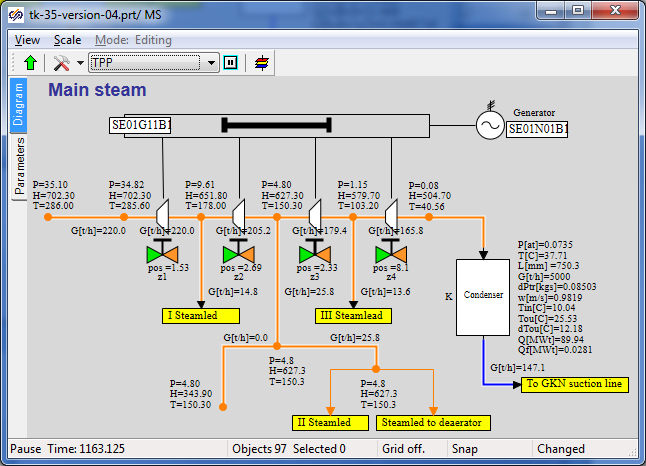


Figure 118. Main Steam System Submodel, Stationary State

Hereon we restrict ourselves with obtaining the nominal – it was important to return the diagram to a stable stationary position. To obtain the nominal state now we have to change conditions in boundary nodes, as well in other points of the diagram – it is not advisably since in the following steps of integration we will have anyway to connect boundaries with links between submodels. Besides, the deaerator at the time is receiving insufficient volume of condensate (from heaters PHP-2 and PHP-3) and its level is gradually reduced, while level in all heaters is artificially maintained due to condensate extraction set equal to steam supply rate. All this points will be corrected at the next stages and “real” nominal state will be able to be generated.

## Connection of Intermediate Circuit

### Model Description

In this step we will deal with connection of two network water heaters for intermediate circuit (PS-450 (1) and PS-450 (2)). Connection of peak heater will be reserved for self-paced training.

Connection points will be as follows (totally 2 (two) ones):

* extraction to PS-450 (1) will be added to the second steam extraction – steam will be supplied to the top volume of network water heater;
* extraction to PS-450 (2) will be added to the second steam extraction – steam will be supplied to the top volume of peak heater.

Heaters will be briefly referred to as PS-1 and PS-2.

Model of intermediate circuit (heaters) will be placed in a separate TPP submodel, as well as previous STP model components.

### STP Model File, Version 05

Open the project, version 04, (“**C:\KTZ\Turbine\ТK-35-version-04.prt**”) and save it in a new file named as **“C:\KTZ\Turbine\ТK-35-version-05.prt**”.

Arrange a **“TPP submodel”** element on the diagram. Change properties of the new page to make their appearance look like Figure 119. Set **“IC”** row as the new page name.

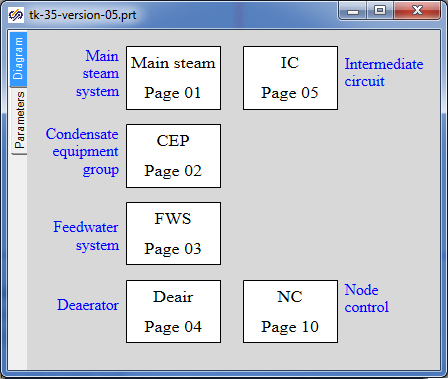


Figure 119. Addition of Intermediate Circuit Submodel Page

### Global Parameters

Two global variables will be required to ensure operability of each intermediate circuit heater model. Add those to the common list of variables (copy them from the heater projects), see Figure 120.

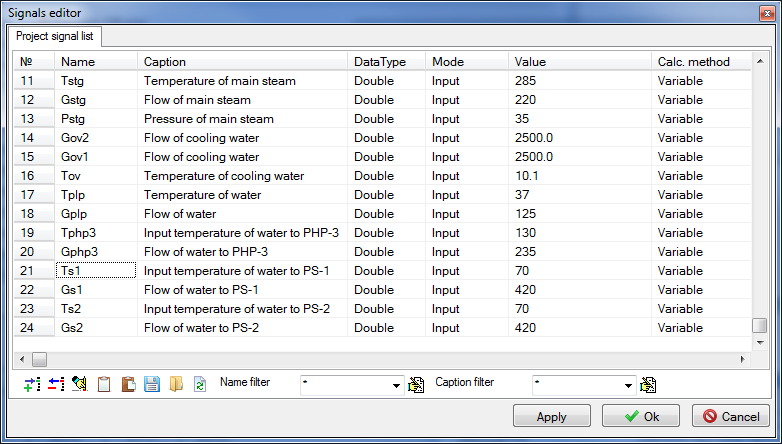


Figure 120. Global Project Signals

We have renamed the variables by adding indexes **“c1”** and **“c2”**. The variables can be saved under any other peculiar names for user-friendliness.

Values for temperature: 70 °С for flow: 420 t/h.

### Structure of Connection of Intermediate Circuit Heaters

Now, copy twice the PS-450 heater submodel from a related project onto the **“IC”** page; place the text for buttons to change boundary conditions and the buttons themselves on another TTP page. Introduce changes into the buttons (and a code for those) since names of global signals have been changed – now we will start to connect the heater models with the STP model.

The both heaters will be connected to the second steam extraction; at the same time we will redistribute two internal TPP nodes and channels among the nodes: the heaters will be connected to the nodes BETWEEN the flowing part point and extraction-to-deaerator point, see Figure 121. Delete boundary node G and replace that with internal TPP node. Thus, we have created a model, in which extraction to DEAER and to PV-280-1 is implemented by “cutting-in” into pipes laid to the heaters.

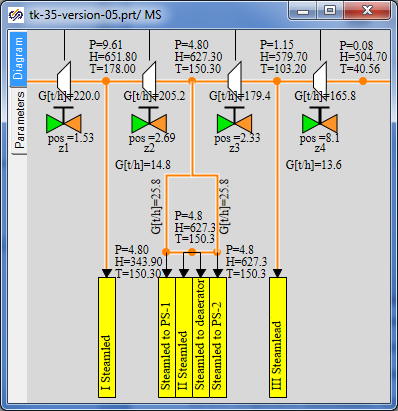


Figure 121. Extractions to Intermediate Circuit Heaters

Rotate the extractions by 90 degrees to conveniently arrange those on the diagram.

Now, add required **“From TPP memory”** blocks on the intermediate circuit submodel page. Note that we have connected the ports to nodes on the main steam system diagram, thus, new TPP nodes are not needed on the heaters diagram, see Figure 122.

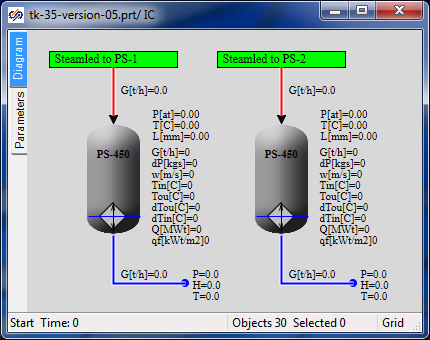


Figure 122. Steam Supply to Intermediate Circuit Heaters

### Display of Parameters in Diagram Window

All relevant parameters have been already displayed in the diagram window.

### Properties of Model Elements

Parameters of about all model elements have been correctly set (parameters of new elements were set as new elements were added).

Change boundary nodes G in the heaters, if required, to set the flow equal to the flow in “own” steam supply channel.

Change other boundary nodes due to a change of names of global project signals for the heaters (in case it occurs SimInTech will deliver an error message).

### Stationary State of Model with Connected Heaters

Launch the diagram for calculation, wait for stationarity is obtained, then analyze and see what its difference from the previous state is.

If the condenser is emptied, close the gate on the head of condensate pumps up to “3...3.5%” value.

In our case the following stationarity for the heaters has been obtained, see Figure 123.

Since we have made minor changes as compared with the previous version, then the stationary state in other parts of the model is similar to the previous stationarity.

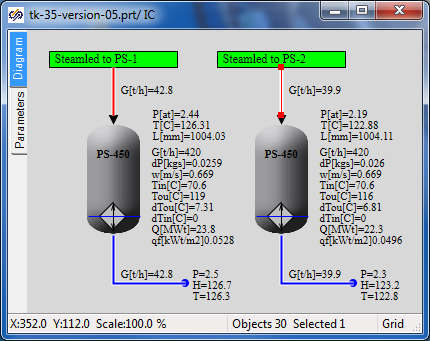


Figure 123. Stationary State of Intermediate Circuit Heaters

Peak heater model is connected to the first steam extraction similarly to the model of network water heaters. You can independently complete the model and make similar manipulations with the peak heater.

## Completion of Deaerator and Heater Models

### Model Description

In this step we will complete the STP model in terms of heating steam motion and its condensate supply to the deaerator from all heaters.

Connection points between submodels of the heaters and deaerator will be as follows (totally 5 (five) ones):

* heating steam condensate will be drained from PLP-1 into the deaerator middle volume;
* heating steam condensate will be delivered from PHP-3 into the PHP-2 middle volume;
* heating steam condensate will be drained from PHP-2 into the deaerator middle volume[[1]](#footnote-1);
* steam condensate will be drained from PS-450 (1) into the deaerator middle volume[[2]](#footnote-2);
* steam condensate will be drained from PS-450 (2) into the deaerator middle volume.

When transferring between submodels within the same page (between PHP-3 and PHP-2) we will not use **“To memory”** and **“From memory”** blocks but arrange the connection directly. All other connections will be arranged via the variable transmitting mechanism in the TPP memory.

Some transfers we are still preparing for connection but will not actually connect those since to debug the nominal state we will have to maintain levels in the heaters, but there are no level controllers yet, i.e., we still cannot “squeeze” the flow in some lines.

### STP Model File, Version 06

Open the project, version 05, (“**C:\KTZ\Turbine\ТK-35-version-05.prt**”) and save it in a new file named as **“C:\KTZ\Turbine\ТK-35-version-06.prt**”.

Remember that there is one boundary condition remained in the deaerator mode, for which we have set a zero flow and have not used that. That will be used for connecting condensate drainage from each heater.

### Global Parameters

Although we do not yet add new submodels we will need three new global signals to arrange connections between the deaerator and three heaters: “PS-1”, “PS-2” and “PHP-2”. In these connections we will “squeeze” the enthalpy of water supplied to the deaerator middle volume to avoid fluctuations in the diagram since in this step there are no level controllers in the heaters and complete connection with the deaerator cannot be enabled.

Set three new global signals: “Hphp2”, “Hps1” and “Hps2”, see Figure 124.

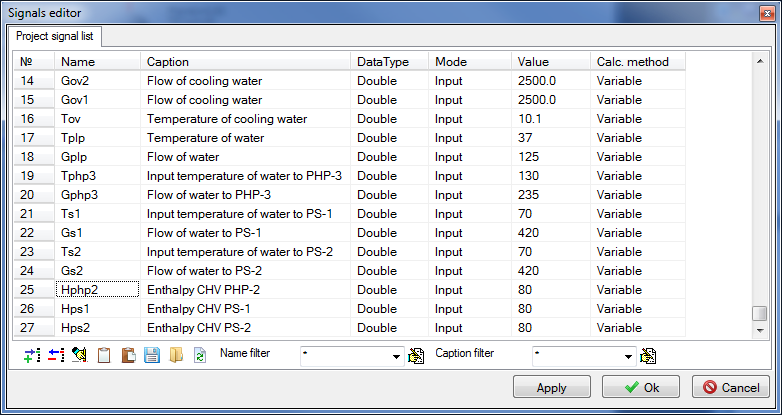


Figure 124. Connection of PHP-2, PS-1 and PS-2 with Deaerator, Global Signals

### Structure of Connection of Heaters to Deaerator

Let us start with the intermediate circuit – we will need to set a TPP control valve on the condensate outlet channel of each heater, add one more internal node and channel and set a gate on the latter (let the controller and gate still be 100% open).

Then set another boundary condition G, set the same parameters in that as in boundary node G connected to the heater and connect that to **“To TPP memory”** block, see Figure 125.

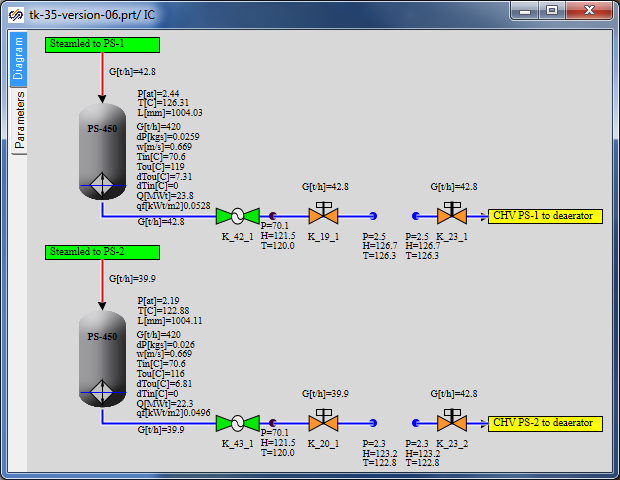


Figure 125. Connection of PS-1 and PS-2 to Deaerator, Addition of Level Controllers

Controller names: “K\_42\_1”, “K\_43\_1”, gate names: “K\_19\_1”, “K\_20\_1”, “K\_23\_1” and “K\_23\_2”. Internal connector node: copy the node between PHP-2 and PHP-3. All new TPP channels shall be created by copying drain channels from the heaters. Thus, we are setting properties of new elements similar to already existing elements – it is quicker than if we manually set properties for every new block on the diagram.

Arrange response “From TPP memory” blocks in the deaerator submodel and connect those to the hole in the middle deaerator volume, see Figure 126.

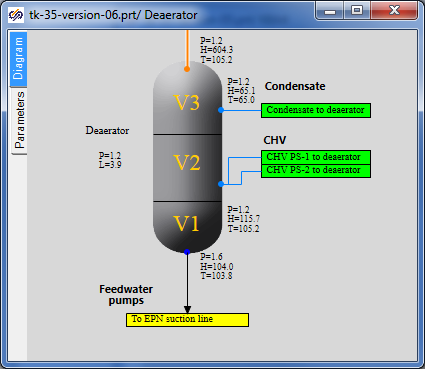


Figure 126. Connection of Intermediate Circuit Heater to Deaerator

To set the water flow and enthalpy in the new boundary node in the process of calculation equal to dynamical values of parameters in the neighboring boundary node G, those shall be assigned with values of “-nodeg6.g” and “Hps1” type, where node6 name is the name of the neighboring node G. Do it for every new boundary node. Then set the enthalpy in new boundary nodes using global signals “**Hps1**” and “**Hps2**”.

Now proceed to PHP-2 heater. Create yourselves the same connection with the deaerator as it has been done for the intermediate circuit heaters. Controller name: “**K\_33\_1**”, position – 100%, here are no gates, see Figure 127 (position of the controller is specified as 100% – it is not correct).

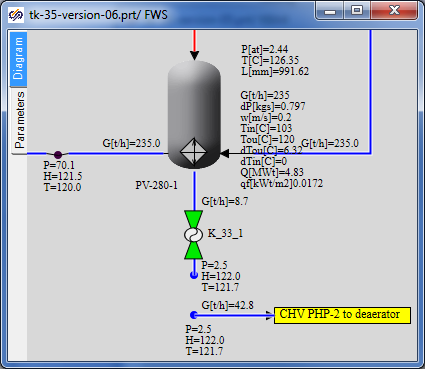


Figure 127. Connection of PHP-2 to Deaerator

Set up “From TPP memory” response block in the deaerator submodel.

In the same way create a link between PLP-1 and the deaerator. The difference here is that there is no controller on the pipeline; the enthalpy will be set the same as in the neighboring boundary node G, i.e., the enthalpy value will be set in the new node as: “nodeg6.h\_”.

Here properties of pipes shall be corrected:

|  |  |
| --- | --- |
| PLP-1 condensate drain channel | Hydraulic diameter: **“0.08”**  Flow area: **“0.005027”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.005”**  Heat transfer surface: **“1.257”**  Length. **“5.0”** |
| Condensate drainage channel from PLP-1 to the deaerator | Hydraulic diameter: **“0.05”**  Flow area: **“0.001963”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.002”**  Heat transfer surface: **“0.7854”**  Length. **“5.0”** |

Example of connection of PLP-1 to the deaerator can be found in Figure 128.

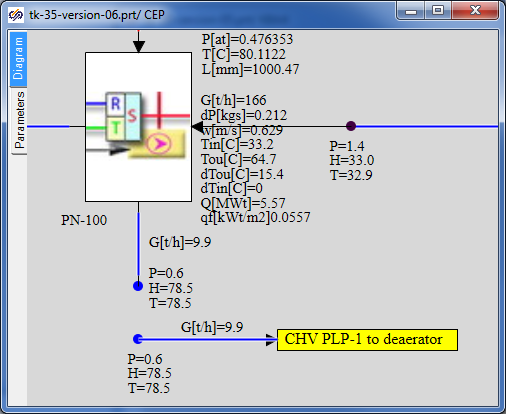


Figure 128. Connection of PLP-1 to Deaerator

The next connection is condensate supply from PHP-3 to PHP-2 middle volume. First, create one more internal node (hole) in the PHP-2 submodel tank by copying the lower hole and changing its property **“Volume No.”** for **“Upper water”**. Place one more **“TPP input port”** here under the name of **“CHS** **from PHP-3”** (**“Heating steam”** port can be copied) and connect that to the new hole as shown in the Figure 129.

Now a new input port has appeared in the PHP-2 submodel.

On copying the existing channel and boundary node, connect the new TPP channel with boundary node G to that (now outside the PHP-2 submodel).

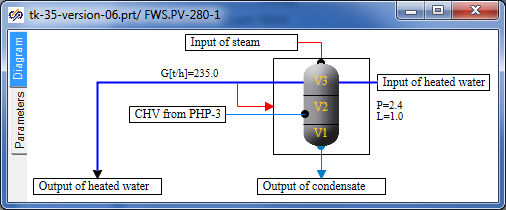


Figure 129. Addition of New Hole to PHP-2

Set controller **“K-34-1”** on the condensate drain channel to 100%. Set the flow and enthalpy in the new boundary node G according to the neighboring node G. Set also check valve **“K\_63\_1”** on the new TPP channel (copy the valve from the pumps model and rename that).

The feed water submodel shall appear like it is depicted in Figure 130.

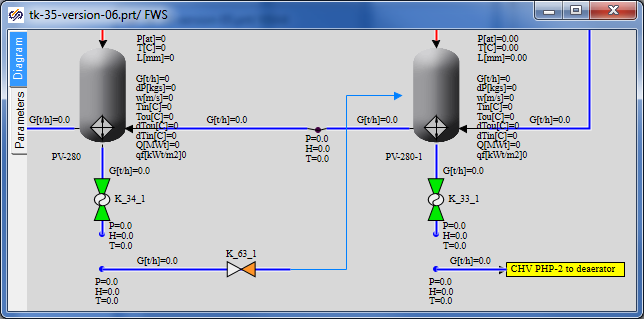


Figure 130. PHP-3 and PHP-2 Connection

After connection of PHP-3 and PHP-2, the flow of condensate supplied from PHP-2 to the deaerator shall be corrected since PHP-3 condensate is now also drained to PHP-2. Set the flow in boundary node G as equal to the sum of two flows **“**-(ch34.g+ch51.g)”.

Now, if you have accomplished the job correctly we can proceed with launching the system for calculation and to analysis of obtained results, detection of a new stationary state and correction of the diagram.

### Display of Parameters in Diagram Window

All relevant parameters have been already displayed in the diagram window.

### Properties of Model Elements

Parameters of about all model elements have been correctly set (parameters of new elements were set as new elements were added). Check, to be on the safe side, all boundary nodes G using the “Find equivalent” menu item to make sure that the flow is correctly set where it shall be set. If SimInTech delivers error messages rectify the errors according to the situation.

### Stationarity with Fully Connected Deaerator

Launch the diagram for calculation, wait for stationarity is obtained then analyze and see what new things have appeared and what the difference from the previous state is. In our case immediately after adding these five new links the stationary state (after 1000 seconds of calculation) was obtained as shown in Figure 131.

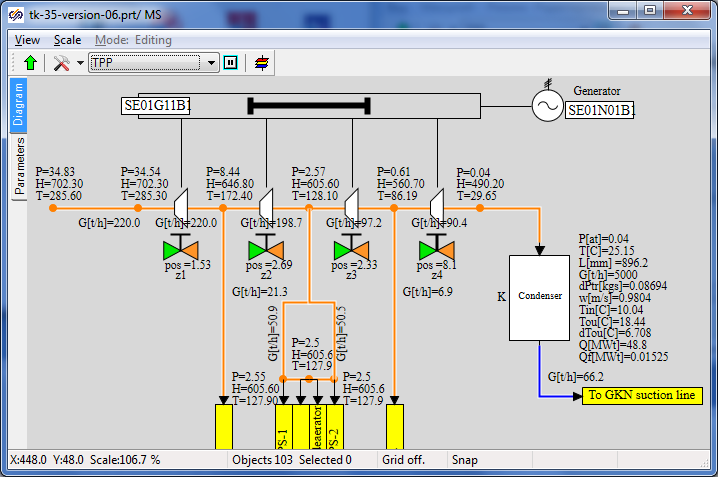


Figure 131. Stationary State after Connecting CHS to Deaerator

## Completion of Flowing Part Model

### Model Description

In this step we will complete the flowing part model in terms of throttle-humidifier device and steam supply to the turbine: add valves **“P\_1\_1”**, **“P\_3\_1”**, **“P\_3\_2”**, **“P\_3\_3”** and create a steam output to the condenser.

There will be 1 (one) connection point between the main steam system and condenser:

* from new valves **“P\_3\_2”** and **“P\_3\_3”** steam will be removed to the condenser.

### STP Model File, Version 07

Open the project, version 06, (“**C:\KTZ\Turbine\ТK-35-version-06.prt**”) and save it in a new file named as “**C:\KTZ\Turbine\ТK-35-version-07.prt**”. All changes in version 07 will be done on the main steam system page.

### Global Parameters

No new equipment will be added in version 07, thus, new global parameters (signals) are not required.

### Structure of Model of Steam Supply to STP

On the **“MSS”** page we have changed a part of the model between boundary node G and the first extraction. We will need to arrange another boundary condition here (type R instead of G currently existing), as well as 4 new gates: one “TPP control gate” and three “TPP pneumatic drive gates”. To that end delete boundary condition G and replace that with a new boundary condition R, shift that to the left.   
Use **“Pstg”** and **“steampt(Pstg\*1e5,Tstg,3)/4182”** as pressure and enthalpy parameters. Add 7 common mode channels (by copying the channel upstream of STP), 4 internal TPP nodes, 4 gates and one **“To TPP memory”** element as per Figure 132. Change the gate names for required ones and the name of **“To TPP memory”** element for **“Draining to condenser”**.

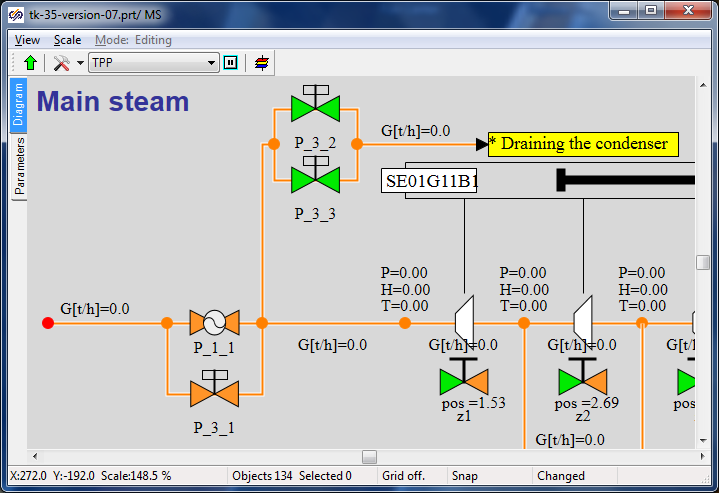


Figure 132. Draining to Condenser Model in Main Steam System

Place the response **“From TPP memory”** element in the condenser submodel and connect that with the internal node as shown in Figure 133.

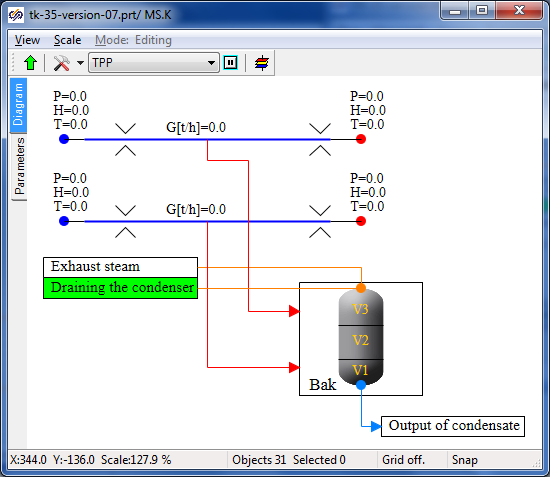


Figure 133. Connection to Condenser

Now parameters of new elements added to the diagram can be set.

### Display of Parameters in Diagram Window

Display flow parameters on the new TPP channels and parameters in nodes (at least in two new TPP nodes).

### Properties of New Model Elements

Change the following parameters of new elements.

|  |  |
| --- | --- |
| Steam supply channels (3 the very first “Common-mode channel” elements) | Direct local resistance: **“0.1”**  Reverse local resistance: **“0.1”**  (All other parameters are the same as the parameters of the channel before the first active element with hydraulic diameter of **“0.5”**). |
| Draining-to-condenser steam channels (4 next “Common-mode channel” elements) | Hydraulic diameter: **“0.25”**  Flow area: **“0.049087”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.01”**  Heat transfer surface: **“3.926”**  Length: **“5.0”** |
| Gate “P\_1\_1” | Position: **“100%”** |
| Gate “P\_3\_1” | Position: **“100%”** |
| Gate “P\_3\_2” | Position: **“0%”** |
| Gate “P\_3\_3” | Position: **“0%”** |
| TPP internal nodes | The same parameters set in TPP by default can be left unchanged here. |
| Boundary node P | Pressure: **“Pstg”**  Enthalpy: **“steampt(Pstg\*1e5,Tstg,3)/4182”** |

### Nominal State

On setting all parameters of new elements, we can launch the task for calculation. In essence, we have produced minor changes but one change was principle – now parameters of steam upstream of STP are set by means of boundary condition P, i.e., pressure and temperature in the steam generator are maintained constant, while the flow shall be adjusted within our STP diagram.

In fact, previous versions, at least version 06 of the STP model, were not absolutely correct since the total water (steam) flow was “squeezed” from the both sides – from the side of steam supply to STP and from the side of heated water outlet from “PHP-3”. In the both cases boundary nodes G were set with a preset **“Gstg”** flow. It is true for the nominal mode but incorrect for dynamical calculations.

Now, if everything has been correctly done in the model then, after launching the diagram for calculation, the steam flow shall reach 220 t/hour within a small period of time after commencement of the calculation. 219.2 t/hour was obtained in our calculation, which corresponds to nominal value with a good degree of accuracy.

## PLP-1 Model Completion

### Model Description

In this step we will complete the low pressure heater model in terms of steam condensate supply to the deaerator – we will add **“EKNS-1”** pump on the condensate drain pipeline.

### STP Model File, Version 08

Open the project, version 07, (“**C:\KTZ\Turbine\ТK-35-version-07.prt**”) and save it in a new file named as **“C:\KTZ\Turbine\ТK-35-version-08.prt**”.

### Global Parameters

No new equipment will be added in version 08, thus, new global parameters (signals) are not required.

### Structure of Model Completion

We will change a part of the model on the **“CEG”** page between **“PLP-1”** and **“PLP-1 CHS to deaerator”** extraction. Here we will need to place two more internal nodes, two channels and gates on each of the channels: **“K\_46\_1”**, **“K\_32\_1”** and **“K\_47\_1”**. We will place an additional “pump without TPP drive element” named as **“EKNS-1”** on the first channel (immediately downstream of PLP-1). Introduce these changes in the diagram, for reference see Figure 134.

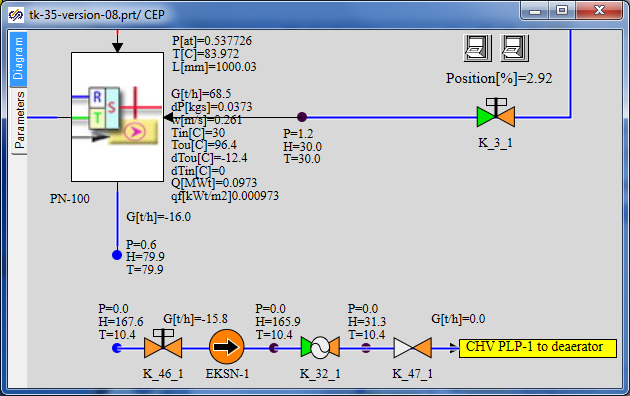


Figure 134. Structure of Introduction of pump EKNS-1 into PLP-1 model

### Display of Parameters in Diagram Window

Display flow parameters on new TPP channels and parameters in two new TPP nodes.

### Properties of New Model Elements

Change the following parameters of new elements.

|  |  |
| --- | --- |
| Channel with a pump (to be divided into two sections) | Number of sections: **“2”**  Hydraulic diameter: **“[0.05,0.05]”**  Flow area: **“[0.0019635,0.0019635]”**  Direct local resistance: **“[1,1]”**  Reverse local resistance: **“[1,1]”**  Wall thickness: **“[0.001,0.001]”**  Heat transfer surface: **“[0.3927,0.3927]”**  Length of section: **“[2.5,2.5]”** |
| Channel with a control valve | Number of sections: **“1”**  Hydraulic diameter: **“0.05”**  Flow area: **“0.0019635”**  Direct local resistance: **“300”**  Reverse local resistance: **“300”**  Wall thickness: **“0.001”**  Heat transfer surface: **“0.3927”**  Length of section: **“5”** |
| Channel with a check valve | Number of sections: **“1”**  Hydraulic diameter: **“0.05”**  Flow area: **“0.0019635”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.001”**  Heat transfer surface: **“0.3927”**  Length of section: **“5”** |
| Gate **“K\_46\_1”** | Position: **“100%”** |
| Control valve **“K\_32\_1”** | Position: **“50%”** |
| Check valve **“K\_47\_1”** | Element No. in channel: **“1”**  Pressure drop, at which the channel is open: **“0.01”**  Open valve resistance factor: **“3”**  Closed valve resistance factor: **“1e8”**  Deadband: **“0.001”** |
| TPP internal nodes | To create the nodes copy the internal node between PLP-1 and the pump group, node parameters will remain unchanged. |
| Pump **“EKNS-1”** | Characteristics of pump: **“EKN\_12-50”** |

### Nominal State

On setting all parameters of new elements, we can launch the task for calculation and make sure that the pump produces a required pressure drop with a preset flow to the deaerator and the system generally works steadily. Possible deviations from the nominal mode will be studied and corrected in the following subsection.

## Completion of Condensate Pumps and Heaters Model

### Model Description

Completion of the group of condensate pumps and heaters consists in the following: we will add recirculation of condensate to the condenser and truly connect CHS drains from all heaters to the deaerator. At the same time we will add level curves for all model tanks (in all heaters) to the model.

Completion and connection points between submodels of the heaters, main steam system and deaerator will be as follows (totally 6 (six) ones):

* add a level controller for PLP-1 between the group of condensate pumps and the heater and create a drain for recirculation to the condenser;
* heating steam condensate from PLP-1 will be drained to the deaerator middle volume (previously prepared link is to be “truly” re-connected);
* heating steam condensate from PHP-3 will be drained to the PHP-2 middle volume (previously prepared link is to be “truly” re-connected);
* CHS from PHP-2 will be drained to the deaerator middle volume (previously prepared link is to be “truly” re-connected);
* steam condensate from PS-450 will be drained to the deaerator middle volume (previously prepared link is to be “truly” re-connected);
* steam condensate from PS-450P will be drained to the deaerator middle volume (previously prepared link is to be “truly” re-connected).

### STP Model File, Version 09

Open the project, version 08, (“**C:\KTZ\Turbine\ТK-35-version-08.prt**”) and save it in a new file named as **“C:\KTZ\Turbine\ТK-35-version-09.prt**”.

### Global Parameters

No new equipment will be added in version 09; thus, new global parameters (signals) are not required. New signals will be required to arrange interaction with some algorithms of automatics – those will be added when required.

### Structure of Model Completion

Let us start with addition of recirculation to the condenser for the group of condensate pumps. Change the model on the “CEG” page in the point where gate **“K\_3\_1”** is located so that it looks like Figure 135. To that end add 5 new internal nodes (copy the node next to gate K\_3\_1), 8 common mode channels (copy the existing channel to copy its hydraulic properties) and 1 **“To TPP memory”** element named as **“CEP recirculation to cond”.** Place relevant gates on the channels.

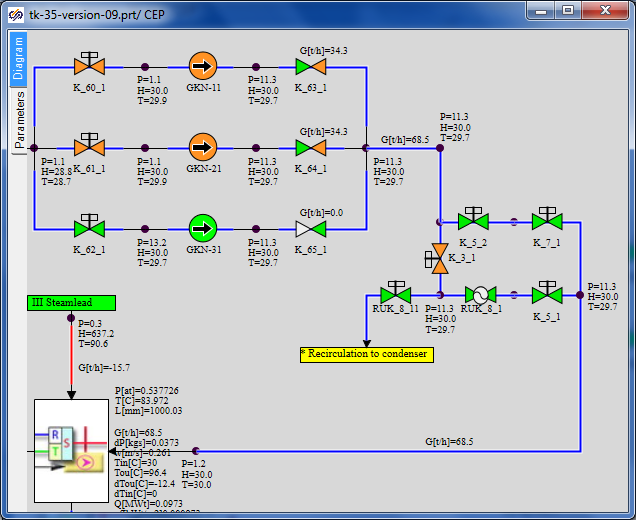


Figure 135. CEP Recirculation to Condenser

Generation of the thermohydraulics part of STP training model is completed at this; the next manual will describe the process of generation of a model of automatic control blocks.

1. Enthalpy of condensate supplied to the deaerator will be set by a constant [↑](#footnote-ref-1)
2. In PS-1 and PS-2 – enthalpy of condensate supplied to the deaerator will be similarly set by a constant. [↑](#footnote-ref-2)