Steam-turbine Plant model development in

SimInTech

Moscow, 2012

Contents

[1 Statement of the Problem 7](#_Toc407108115)

[1.1 Basic Data 7](#_Toc407108116)

[2 Creation of a Thermohydraulics Model for the Flowing Part 8](#_Toc407108117)

[2.1 Creation of a New Thermohydraulics Model 8](#_Toc407108118)

[2.2 Check for Connection to Signal Database 9](#_Toc407108119)

[2.3 Setting the Flowing Part Diagram 12](#_Toc407108120)

[2.4 Adjustment of Calculation Model Parameters and Element Properties 15](#_Toc407108121)

[2.4.1 Boundary Node P 15](#_Toc407108122)

[2.4.2 Global Project Signals 16](#_Toc407108123)

[2.4.3 Boundary Node G 17](#_Toc407108124)

[2.4.4 Nodes G Corresponding to Steam Extractions 18](#_Toc407108125)

[2.4.5 Internal Nodes of Flowing Part Model 21](#_Toc407108126)

[2.4.6 Rotor 22](#_Toc407108127)

[2.4.7 Generator 22](#_Toc407108128)

[2.4.8 Diagram Debugging Elements 22](#_Toc407108129)

[2.4.9 Pipelines 23](#_Toc407108130)

[2.4.10 Gates 25](#_Toc407108131)

[2.4.11 Control of TPP Parameters 26](#_Toc407108132)

[2.5 Rated State 32](#_Toc407108133)

[2.5.1 General Principles for Debugging the Thermohydraulics Diagram 32](#_Toc407108134)

[2.5.2 Methods for Selecting Flowing Part Parameters 32](#_Toc407108135)

[2.5.3 Active Elements, Rotor and Generator 33](#_Toc407108136)

[3 Generation of thermohydraulics condenser model 35](#_Toc407108137)

[3.1 Creation of a new TPP diagram 35](#_Toc407108138)

[3.1.1 New TPP diagram 35](#_Toc407108139)

[3.1.2 Global parameters 35](#_Toc407108140)

[3.2 Main condenser model 36](#_Toc407108141)

[3.2.1 Model description 36](#_Toc407108142)

[3.2.2 Creation of condenser model top level 37](#_Toc407108143)

[3.2.3 Creation of condenser submodel nested level 39](#_Toc407108144)

[3.2.4 New block editor 41](#_Toc407108145)

[3.2.5 Display of parameters in diagram window 51](#_Toc407108146)

[3.2.6 Properties of boundary conditions 52](#_Toc407108147)

[3.2.7 Properties of condenser lines 53](#_Toc407108148)

[3.2.8 Properties of 3V TPP tank and the project as a whole 53](#_Toc407108149)

[3.2.9 Rated state 56](#_Toc407108150)

[4 Creation of feed water heater models 57](#_Toc407108151)

[4.1 Creation of PND-1 model as a basis for all heaters 57](#_Toc407108152)

[4.1.1 New TPP diagrams 57](#_Toc407108153)

[4.1.2 Setting of model PND-1 global parameters 58](#_Toc407108154)

[4.1.3 Setting of PND-1 model structure 59](#_Toc407108155)

[4.1.4 Programming of PND-1 (PN-100) heater submodel 61](#_Toc407108156)

[4.1.5 Display of parameters in diagram window 64](#_Toc407108157)

[4.1.6 Properties of boundary nodes, channels and other elements of PND-1 model 65](#_Toc407108158)

[4.1.7 Parameters of PND-1 calculation 66](#_Toc407108159)

[4.1.8 PND-1 nominal state 67](#_Toc407108160)

[4.2 Creation of PVD-2 model on the basis of PND-1 67](#_Toc407108161)

[4.2.1 Project copying, calculation parameters 67](#_Toc407108162)

[4.2.2 Global parameters 68](#_Toc407108163)

[4.2.3 PVD-2 model structure 68](#_Toc407108164)

[4.2.4 PVD-2 submodel 69](#_Toc407108165)

[4.2.5 Display of parameters in diagram window 70](#_Toc407108166)

[4.2.6 Properties of boundary nodes, channels and other elements of PVD-2 model 70](#_Toc407108167)

[4.2.7 Parameters of PVD-2 calculation 71](#_Toc407108168)

[4.2.8 PVD-2 nominal state 71](#_Toc407108169)

[4.3 Creation of PVD-3 model on the basis of PVD-2 71](#_Toc407108170)

[4.3.1 Project copying, calculation parameters 71](#_Toc407108171)

[4.3.2 Global parameters 72](#_Toc407108172)

[4.3.3 PVD-3 model structure 72](#_Toc407108173)

[4.3.4 PVD-3 submodel 72](#_Toc407108174)

[4.3.5 Display of parameters in diagram window 73](#_Toc407108175)

[4.3.6 Properties of boundary nodes, channels and other elements of PVD-2 model 73](#_Toc407108176)

[4.3.7 Parameters of PVD-2 calculation 73](#_Toc407108177)

[4.3.8 PVD-2 nominal state 73](#_Toc407108178)

[5 Creation of models of intermediate circuit heaters 75](#_Toc407108179)

[5.1 Creation of model of PS-450 network water heater 75](#_Toc407108180)

[5.1.1 Project copying, calculation parameters 75](#_Toc407108181)

[5.1.2 Global parameters 75](#_Toc407108182)

[5.1.3 PS-450 model structure 75](#_Toc407108183)

[5.1.4 PS-450 submodel 76](#_Toc407108184)

[5.1.5 Display of parameters in diagram window 76](#_Toc407108185)

[5.1.6 Properties of boundary nodes, channels and other elements of PS-450 model 76](#_Toc407108186)

[5.1.7 PS-450 calculation parameters 76](#_Toc407108187)

[5.1.8 PS-450 nominal state 77](#_Toc407108188)

[5.2 Creation of model of PS-450P peak heater 77](#_Toc407108189)

[5.2.1 Project copying, calculation parameters 77](#_Toc407108190)

[5.2.2 Global parameters 78](#_Toc407108191)

[5.2.3 PS-450P model structure 78](#_Toc407108192)

[5.2.4 PS-450P submodel 78](#_Toc407108193)

[5.2.5 Display of parameters in diagram window 78](#_Toc407108194)

[5.2.6 Properties of boundary nodes, channels and other elements of PS-450P model 78](#_Toc407108195)

[5.2.7 PS-450P calculation parameters 79](#_Toc407108196)

[5.2.8 PS-450P nominal state 79](#_Toc407108197)

[6 Creation of models of pump blocks 81](#_Toc407108198)

[6.1 Creation of model of condensate pump block 81](#_Toc407108199)

[6.1.1 New TPP diagram 81](#_Toc407108200)

[6.1.2 EKN-150-110 global parameters 81](#_Toc407108201)

[6.1.3 Setting of EKN-150-110 model structure 81](#_Toc407108202)

[6.1.4 Display of parameters in diagram window 83](#_Toc407108203)

[6.1.1 Properties of nodes, channels, pumps and other elements of EKN-150-110 model 84](#_Toc407108204)

[6.1.2 Nominal state of condensate pump model 86](#_Toc407108205)

[6.2 Creation of model of feed pump block 87](#_Toc407108206)

[6.2.1 New TPP diagram 87](#_Toc407108207)

[6.2.2 EPN-150-75 global parameters 87](#_Toc407108208)

[6.2.3 Setting of EPN-150-75 model structure 87](#_Toc407108209)

[6.2.4 Display of parameters in diagram window 89](#_Toc407108210)

[6.2.5 Properties of nodes, channels, pumps and other elements of EPN-150-75 model 89](#_Toc407108211)

[6.2.6 Nominal state of condensate pump model 91](#_Toc407108212)

[7 Creation of deaerator model 92](#_Toc407108213)

[7.1 Description of deaerator model 92](#_Toc407108214)

[7.2 Creation of deaerator model 92](#_Toc407108215)

[7.2.1 Project copying, calculation parameters 92](#_Toc407108216)

[7.2.2 Global parameters of deaerator model 92](#_Toc407108217)

[7.2.3 Structure of deaerator model 92](#_Toc407108218)

[7.2.4 Display of parameters in diagram window 93](#_Toc407108219)

[7.2.5 Properties of deaerator model elements 93](#_Toc407108220)

[7.2.6 Parameters of deaerator calculation 94](#_Toc407108221)

[7.2.7 Nominal state of deaerator 94](#_Toc407108222)

[8 Integration of Models, Generation of Integrated STP Model 96](#_Toc407108223)

[8.1 Main Steam Model Generation 96](#_Toc407108224)

[8.1.1 Model Description 96](#_Toc407108225)

[8.1.2 STP Model File, Version 01 96](#_Toc407108226)

[8.1.3 Global Parameters 97](#_Toc407108227)

[8.1.4 Integration of the Flow Part and Condenser Models, Structure 98](#_Toc407108228)

[8.1.5 Display of Parameters in Diagram Window 99](#_Toc407108229)

[8.1.6 Properties of Main Steam Model Elements 99](#_Toc407108230)

[8.1.7 Nominal State of Main Steam System 100](#_Toc407108231)

[8.1.8 Control of Generator and Turbine Rotor Parameters 100](#_Toc407108232)

[8.2 Connection of the Main Steam System and Condensate Equipment Group 102](#_Toc407108233)

[8.2.1 Model Description 102](#_Toc407108234)

[8.2.2 STP Model File, Version 02 102](#_Toc407108235)

[8.2.3 Global Parameters 103](#_Toc407108236)

[8.2.4 Structure of Connection of Condensate Equipment Group 103](#_Toc407108237)

[8.2.5 Display of Parameters in Diagram Window 107](#_Toc407108238)

[8.2.6 Properties of Model Elements 107](#_Toc407108239)

[8.2.7 System Nominal State 107](#_Toc407108240)

[8.3 Connection of Feedwater System 107](#_Toc407108241)

[8.3.1 Model Description 107](#_Toc407108242)

[8.3.2 STP Model File, Version 03 108](#_Toc407108243)

[8.3.3 Global Parameters 108](#_Toc407108244)

[8.3.4 Structure of Connection of Feedwater System 109](#_Toc407108245)

[8.3.5 Display of Parameters in Diagram Window 111](#_Toc407108246)

[8.3.6 Properties of Model Elements 111](#_Toc407108247)

[8.3.7 System Nominal State 112](#_Toc407108248)

[8.4 Connection of Deaerator 114](#_Toc407108249)

[8.4.1 Model Description 114](#_Toc407108250)

[8.4.2 STP Model File, Version 04 114](#_Toc407108251)

[8.4.3 Global Parameters 115](#_Toc407108252)

[8.4.4 Structure of Connection of Deaerator Submodel 115](#_Toc407108253)

[8.4.5 Display of Parameters in Diagram Window 117](#_Toc407108254)

[8.4.6 Properties of Model Elements 117](#_Toc407108255)

[8.4.7 Stationary State of the Model with Connected Deaerator Model 117](#_Toc407108256)

[8.5 Connection of Intermediate Circuit 118](#_Toc407108257)

[8.5.1 Model Description 118](#_Toc407108258)

[8.5.2 STP Model File, Version 05 119](#_Toc407108259)

[8.5.3 Global Parameters 119](#_Toc407108260)

[8.5.4 Structure of Connection of Intermediate Circuit Heaters 120](#_Toc407108261)

[8.5.5 Display of Parameters in Diagram Window 121](#_Toc407108262)

[8.5.6 Properties of Model Elements 121](#_Toc407108263)

[8.5.7 Stationary State of Model with Connected Heaters 122](#_Toc407108264)

[8.6 Completion of Deaerator and Heater Models 122](#_Toc407108265)

[8.6.1 Model Description 122](#_Toc407108266)

[8.6.2 STP Model File, Version 06 123](#_Toc407108267)

[8.6.3 Global Parameters 123](#_Toc407108268)

[8.6.4 Structure of Connection of Heaters to Deaerator 124](#_Toc407108269)

[8.6.5 Display of Parameters in Diagram Window 128](#_Toc407108270)

[8.6.6 Properties of Model Elements 128](#_Toc407108271)

[8.6.7 Stationarity with Fully Connected Deaerator 128](#_Toc407108272)

[8.7 Completion of Flowing Part Model 129](#_Toc407108273)

[8.7.1 Model Description 129](#_Toc407108274)

[8.7.2 STP Model File, Version 07 129](#_Toc407108275)

[8.7.3 Global Parameters 129](#_Toc407108276)

[8.7.4 Structure of Model of Steam Supply to STP 129](#_Toc407108277)

[8.7.5 Display of Parameters in Diagram Window 131](#_Toc407108278)

[8.7.6 Properties of New Model Elements 131](#_Toc407108279)

[8.7.7 Nominal State 132](#_Toc407108280)

[8.8 PLP-1 Model Completion 132](#_Toc407108281)

[8.8.1 Model Description 132](#_Toc407108282)

[8.8.2 STP Model File, Version 08 132](#_Toc407108283)

[8.8.3 Global Parameters 132](#_Toc407108284)

[8.8.4 Structure of Model Completion 132](#_Toc407108285)

[8.8.5 Display of Parameters in Diagram Window 133](#_Toc407108286)

[8.8.6 Properties of New Model Elements 133](#_Toc407108287)

[8.8.7 Nominal State 134](#_Toc407108288)

[8.9 Completion of Condensate Pumps and Heaters Model 134](#_Toc407108289)

[8.9.1 Model Description 134](#_Toc407108290)

[8.9.2 STP Model File, Version 09 134](#_Toc407108291)

[8.9.3 Global Parameters 135](#_Toc407108292)

[8.9.4 Structure of Model Completion 135](#_Toc407108293)

Abbreviations

Deaerator – means here “de-air equipment”

EKN – electro-condensate pump

EPN – electro-feed-water pump

MCP – main condensate pump

PND – lower pressure heater (LPH)

PS – network heater

PVD – higher pressure heater (HPH)

SimInTech – Simulation In Technical Systems

STP – steam-turbine plant (model of)

TPP – Thermal Power Plant (name of the thermodynamical code, which is used)

# Statement of the Problem

## Basic Data

Specifications of JSC Kaluga Turbine Works have been used herein as basic data: “Steam Turbine Plant ТК-35/38-3,4. ИРЕЦ 624121.001ТУ”.

In accordance with the basic data the following rated parameters of steam turbine plant are accepted for modeling:

Steam pressure upstream of the turbine: 35 kgf/cm2.

Flow of main steam for the turbine: 220 t/h = 61.111 kg/s

Steam temperature upstream of steam turbine plant shut-off valves: 285 °С

Heating steam flow in the 1st extraction (to PHP No. 3): 10 t/h

Heating steam flow in the 12nd extraction (to PHP No. 2): 13 t/h

Heating steam flow in the 3rd extraction (to PLP No. 1): 18.4 t/h

Steam flow from adjustable turbine heat extraction: 66.6 t/h

Steam pressure in condenser: 0.005 MPa = 0.051 kgf/cm2

Condensate temperature: 32 °С

Steam pressure in the 1st extraction: 0.913 MPa = 9.2 kgf/cm2, steam temperature: +170 °С

Steam pressure in the 2nd extraction: 0.35 MPa = 3.6 kgf/cm2, steam temperature: +140 °С

Steam pressure in the 3rd extraction: 0.094 MPa = 0.96 kgf/cm2, steam temperature: +91 °С

Steam flow from adjustable heat extraction at the intermediate circuit heater:   
52 t/h = 14.444 kg/s

Note: main initial data are given herein. As the model is created other equipment parameters and characteristics will be added.

# Creation of a Thermohydraulics Model for the Flowing Part

## Creation of a New Thermohydraulics Model

To create a thermohydraulics diagram in SimInTech proceed as follows:

1. Select **“New Project”** button in the toolbar.

Select “TPP Diagram” item from a pop-up menu (see Figure 1. New Thermohydraulics Project Creation Menu

1. ).

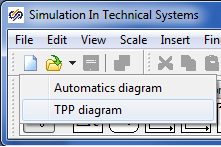


Figure 1. New Thermohydraulics Project Creation Menu

Then a new diagram window will be opened, in which a thermohydraulics block diagram will be generated (see Figure 2. Thermohydraulics Diagram Creation Window

).



Figure 2. Thermohydraulics Diagram Creation Window

To proceed with the work save the given diagram in a file under a new name (e.g., “Flowing part”). To that end, perform the following operations:

1. Select item **“File”** in the main menu and then select **“Save project as...”** item from the pop-up list.
2. Use the standard “save file” dialog to select a new name and catalog for saving. “C:\KTZ\Turbine\Flowing part\Flowing part.prt”.

When the file has been saved, its name and full path are displayed in the diagram window title (see Figure 4. Diagram Window with New and Saved Projects

).

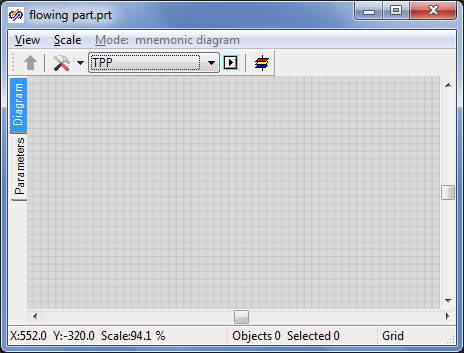


Figure 4. Diagram Window with New and Saved Projects

## Check for Connection to Signal Database

Then, after completion of debugging of all parts of the turbine model, we will need to combine separate parts in an integral project (besides, the model of automatics system shall be engaged in the hydraulic calculation). To ensure a common operation of several calculation codes it is necessary that those use the same signal database. Thus, it would be advisable to make provision for a database under the same name and uniform signal names to be used for all projects. By default, database with file name “tpp.db” is used in TPP project. The file is located in the current directory of the project. This file name is quite good for us and we will use it.

To check the connection of the database in a newly generated thermohydraulics project switch over the software system to the developer mode, for which purpose select “File” item in the main program menu and then “Parameters” sub-item. Go to “View” tab in the displayed “Parameters” dialog window and tick “Developer mode” option (see Figure 5. “Parameters” Dialog Window for Activation of Developer Mode

).



Figure 5. “Parameters” Dialog Window for Activation of Developer Mode

Signal database is connected to the thermohydraulics diagram in the following manner:

1. Press button **“Simulation properties”** in the diagram window:

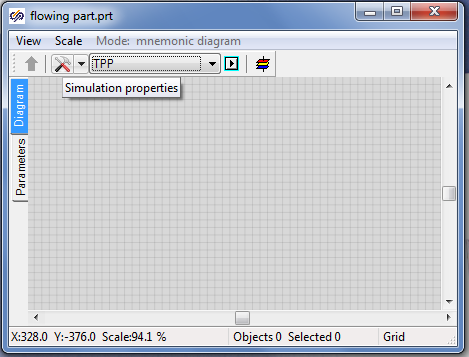


Figure 6. Simulation Properties Access Button

Go to “Settings” tab in the popped up setting dialog (see Figure 7. Project Database Setting Tab

1. ).
2. Enter the following text in **“Project database module”** edit bar: “$(Root)\sdb.dll” (to be entered without quotation marks; sdb.dll is for the name of dynamic library of database program module).

Enter a random file name for saving the database in “Project database name” bar. In our case we just make sure that everything is correctly filled, and leave the default file name (“tpp\_eng.db”, see Figure 7. Project Database Setting Tab

1. ).

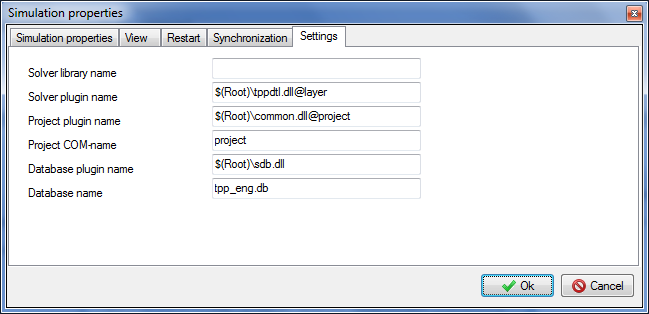


Figure 7. Project Database Setting Tab

1. Close the dialog window by pressing **“Ok”** button (see Figure 7. Project Database Setting Tab
2. ).

## Setting the Flowing Part Diagram

Blocks located in **“TPP process blocks”** tab of the blocks template are used to create the diagram. (see Figure 8. Thermohydraulics Blocks Pop-up List

).

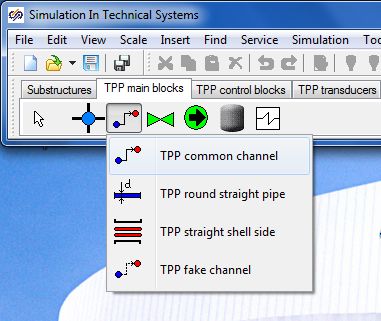


Figure 8. Thermohydraulics Blocks Pop-up List

For modeling the flowing part we will need 9 channels of common type, 5 internal TPP nodes, 4 boundary nodes of G type, one boundary node of P type and some other elements…

The sequential operations are executed:

1. Arrange the following design thermohydraulics blocks in the diagram window:

– “Boundary node P” (in the right-hand part of the calculation diagram).

– “Boundary node G” (at the left on the calculation diagram).

– “Internal node” (4 nodes in series).

– “Boundary node G” (three nodes at the bottom under TPP internal nodes).

– “Internal node” (one more, next to the internal node of P type).

– “Common type channel” (9 pipes in series).

The pipes are advisably to be arranged separately, do not connect those to the nodes. This will allow those to be more accurately connected to the nodes and, thus, no “problem” points will occur (sometimes a gap between a node and a pipe cannot be seen). After completion of all arrangements, we will obtain a picture the same as the figure below:

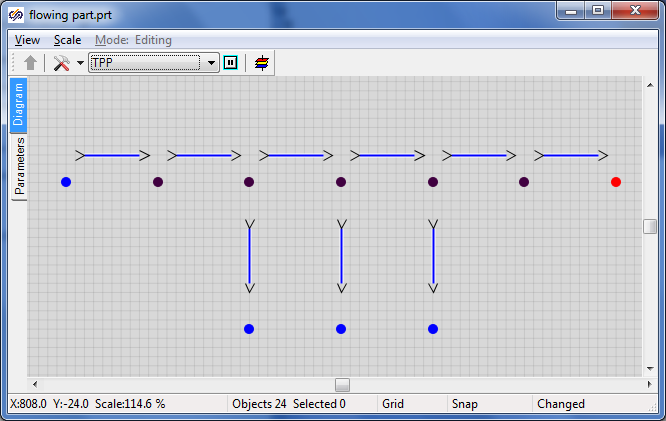


Figure 9. Initial Step for Setting the Flowing Part Thermohydraulics Diagram

Link successively the elements so that “Common type channels” elements generate one hydraulic line with internal nodes. In our model boundary nodes “G” will define the steam flow at the inlet to the flowing part and in steam extractions, while boundary node “P” – the pressure at the boundary with the condenser (see Figure 10. Generation of a Common Hydraulic Line

1. ).
2. Place one **“Remotely operated manual gate”** element of TPP code in each one of the four intermediate (not connected to boundary condition) channels. Change the name of each gate for a new one: z1, z2, z3 and z4, correspondingly.
3. Shift the gates just below the channels on the diagram for convenience.

Place “Active TPP element” element from “Turbo-pump plant elements” tab on each one of these channels (with manual gates) (see Figure 11. Addition of Gates and Turbine Active Elements

1. ).

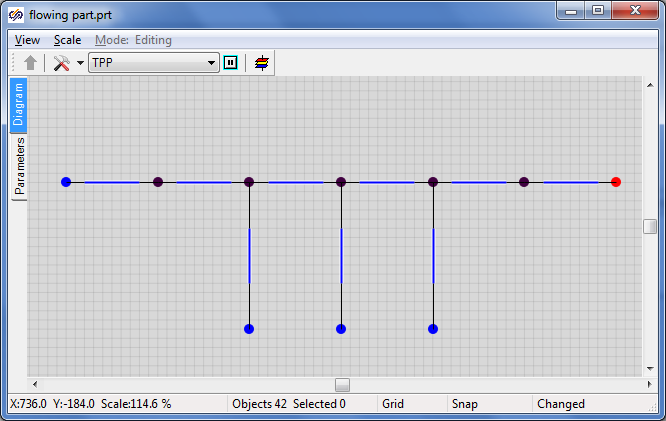


Figure 10. Generation of a Common Hydraulic Line

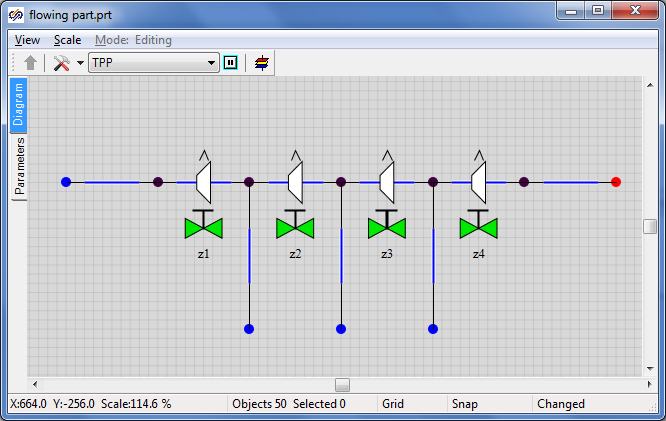


Figure 11. Addition of Gates and Turbine Active Elements

1. Add **“TPP rotor”** element and change the value of its **“Number of mechanical ports”** property for **“5”**. Place the fifth port from the right in “Ports” tab (the others will remain at the bottom).
2. Link the rotor ports with four bottom extraction ports. Set the **“Show the frame”** rotor property to **“Yes”**.
3. Place the **“TPP generator”** element at the right from the rotor. Link the generator to the rotor with a junction line. The diagram can be executed in a prettier manner – work with that will be more pleasant and modeling quality will be enhanced. For example, the rotor size can be increased considering the extractions.

As a result, thermohydraulics model diagram shall resemble the Figure 12. Completion of Setting of the Turbine Flowing Part Hydraulics Diagram

:

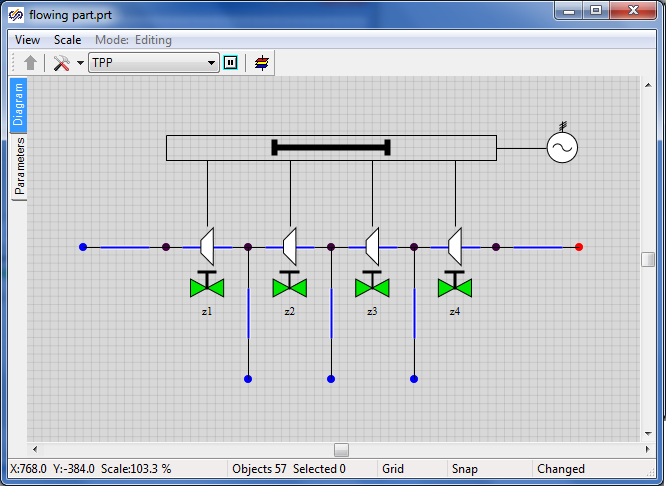


Figure 12. Completion of Setting of the Turbine Flowing Part Hydraulics Diagram

## Adjustment of Calculation Model Parameters and Element Properties

### Boundary Node P

To ensure correct calculation of thermohydraulics model, properties of each diagram element shall be set in the **“Properties”** dialog window next to each element. Dialog window for **“Boundary node P”** object is presented below (see Figure 13. “Properties” Dialog Window for Boundary Node P

).

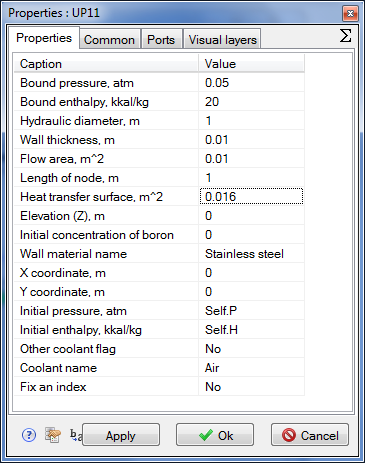


Figure 13. “Properties” Dialog Window for Boundary Node P

Set new values for the properties in the boundary node of **“P”** type:

Pressure: 0.05

Enthalpy: 20

Flow area: 1

Heat transfer surface: 1

Leave the other properties by default, i.e., unchanged. Thus, we have set the constant rated value in the turbine condenser, which will be then relied upon in the process of debugging of the flowing part model.

### Global Project Signals

To set global project constants the signal mechanism will be used. Go to **“Graphics”** → “Signals…” item via the main SimInTech menu.

In a window activated you will see ten signals (constants) set by default and used in scripts of a number of standard TPP elements (see Figure 14. “Project Signals Editor” Dialog Window with Signals by Default

). We will have to set three new constants (signals) – pressure, flow and temperature for steam supplied to the turbine: “Pstg”, “Gstg” and “Tstg”. Set their values as per Figure 15. “Project Signals Editor” Dialog Window with Three New Signals

:

Pstg – Pressure of main steam upstream of the turbine – Real – Input – 35

Gstg – Flow of main steam for the turbine – Real – Input – 220

Tstg – Temperature of main steam upstream of STP – Real – Input – 285

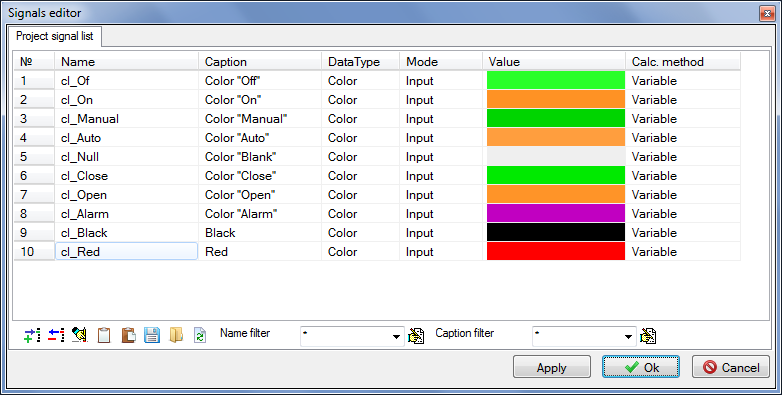


Figure 14. “Project Signals Editor” Dialog Window with Signals by Default

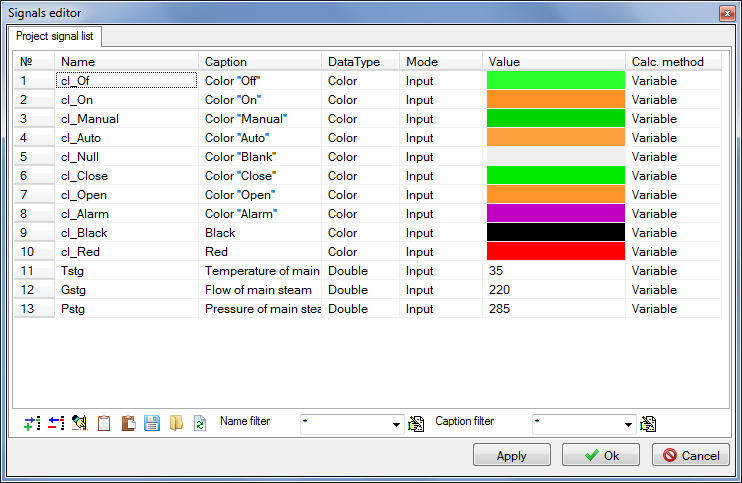


Figure 15. “Project Signals Editor” Dialog Window with Three New Signals

### Boundary Node G

Set the following values of properties in the very first (to the left) boundary node **“G”** (see Figure 16. “Properties” Dialog Window for Boundary Node G

):

Flow: **“Gstg/3.6”** – conversion from t/h to kg/s.

Enthalpy: “steampt(Pstg\*1e5,Tstg,3)/4182” – kcal is obtained from pressure and temperature.

Hydraulic diameter: 1

Flow area: 1

Length of node: 1

Heat transfer surface: 1

Initial pressure: “Pstg” – is to be taken from the signal list.

Initial enthalpy: “Self.H” – enthalpy has been calculated earlier.

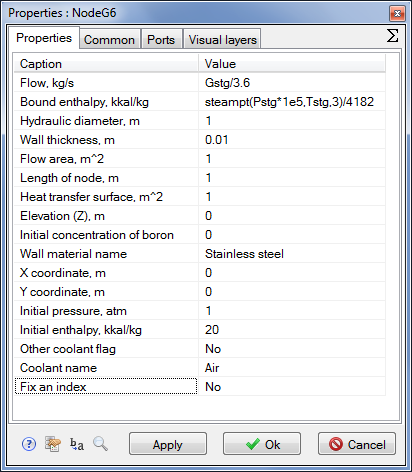


Figure 16. “Properties” Dialog Window for Boundary Node G

### Nodes G Corresponding to Steam Extractions

Set the following properties in nodes G corresponding to steam extractions (see also Figure 17. “Properties” Dialog Window for Boundary Node G (Steam Extraction I)

, Figure 18. “Properties” Dialog Window for Boundary Node G (Steam Extraction II)

and Figure 19. “Properties” Dialog Window for Boundary Node G (Steam Extraction III)

):

**Steam extraction I:**

Flow: **“**-18.4/3.6**”** – conversion from t/h to kg/s.

Enthalpy: “steamps(9.2e5,3)/4182” – steam enthalpy in kcal on saturation line.

Hydraulic diameter: “1”

Flow area: “1”

Length of node: “1”

Heat transfer surface: “1”

Initial pressure: **“9.2”** – steam pressure in the first extraction.

Initial enthalpy: **“Self.H”** – from the node properties (enthalpy has been calculated earlier).

Steam extraction II:

Flow: **“**-66.6/3.6**”** – as per initial data, conversion from t/h to kg/s.

Enthalpy: “steamps(3.64e5,3)/4182” – steam enthalpy in kcal on saturation line.

Hydraulic diameter: “1”

Flow area: “1”

Length of node: “1”

Heat transfer surface: “1”

Initial pressure: **“3.64”** – steam pressure in the second extraction.

Initial enthalpy: **“Self.H”** – from the node properties (enthalpy has been calculated earlier).

Steam extraction III:

Flow: **“**-10.0/3.6**”** – conversion from t/h to kg/s.

Enthalpy: “steamps(0.96e5,3)/4182” – steam enthalpy in kcal on saturation line.

Hydraulic diameter: “1”

Flow area: “1”

Length of node: “1”

Heat transfer surface: “1”

Initial pressure: **“0.96”** – steam pressure in the third extraction.

Initial enthalpy: **“Self.H”** – from the node properties (enthalpy has been calculated earlier).

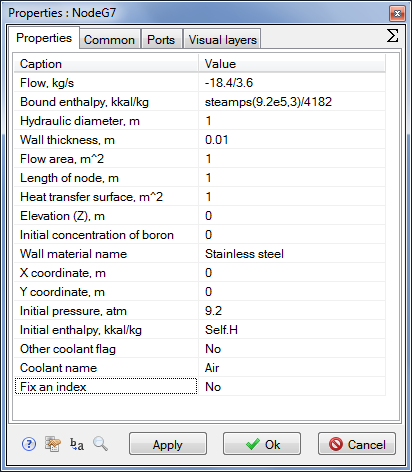


Figure 17. “Properties” Dialog Window for Boundary Node G (Steam Extraction I)

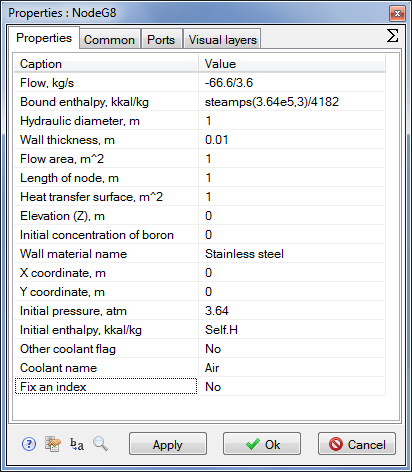


Figure 18. “Properties” Dialog Window for Boundary Node G (Steam Extraction II)

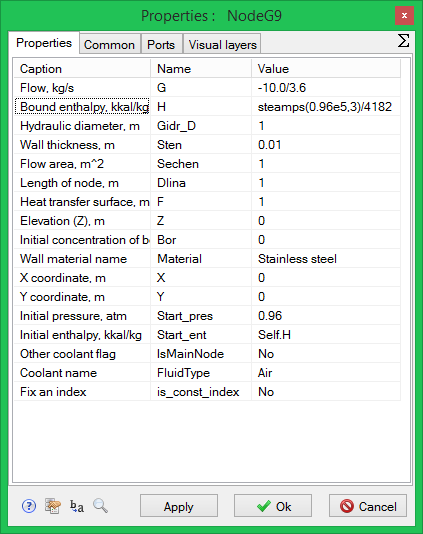


Figure 19. “Properties” Dialog Window for Boundary Node G (Steam Extraction III)

Thus, we have set all necessary properties in all boundary nodes of the task. Now let us go to setting properties in internal TPP nodes.

### Internal Nodes of Flowing Part Model

Internal TPP node has a set of properties similar to the one for boundary nodes. Let us set the following values for every internal node from left to right:

|  |  |
| --- | --- |
| Internal node No. 1 | Initial enthalpy: **“702.25”**  Hydraulic diameter: **“1.2”**  Flow area: **“4.521”**  Length of node: **“0.2”**  Heat transfer surface: **“1.508”**  Initial pressure: **“10.47”** |
| Internal Node No. 2 (to steam extraction I) | Initial enthalpy: **“676.06”**  Hydraulic diameter: **“0.1313”**  Flow area: **“0.7904”**  Length of node: **“0.1”**  Heat transfer surface: **“0.041”**  Initial pressure: **“5.879”** |
| Internal Node No. 3 (to steam extraction II) | Initial enthalpy: **“651.36”**  Hydraulic diameter: **“0.2”**  Flow area: **“0.9126”**  Length of node: **“0.1”**  Heat transfer surface: **“0.0628”**  Initial pressure: **“3.162”** |
| Internal Node No. 4 (to steam extraction III) | Initial enthalpy: **“532”**  Hydraulic diameter: **“0.4367”**  Flow area: **“0.9766”**  Length of node: **“0.1”**  Heat transfer surface: **“0.1371”**  Initial pressure: **“0.867”** |
| Internal node No. 5 | Initial enthalpy: **“532”**  Hydraulic diameter: **“0.4367”**  Flow area: **“0.9766”**  Length of node: **“0.1”**  Heat transfer surface: **“0.1371”**  Initial pressure: **“0.867”** |

### Rotor

Let us set the following values in the rotor properties:

Rotor inertia moment: “625878.0”

Rated rotation speed: “50.0”

Initial rotation speed: “50.0”

Dependency table for moment of resistance: “[[28000,28000],[28000,28000]]”

Note that the fractional part of the number is separated from the integral part with a dot, while groups are set by means of commas.

### Generator

Let us set the value in the generator properties:

Rotation speed: “50.0”

### Diagram Debugging Elements

In order to simplify and accelerate the debugging process for the flowing part diagram we will need to arrange 8 elements more on the diagram: four buttons and four text elements for controlling boundary conditions in terms of steam temperature and flow at STP.

In the process of calculation we will increment and decrement the values of global constants (variables) “Gstg” and **“**Tstg”. Two buttons will be used to increment the variables and two others to decrement those.

Arrange four buttons “Button” and four text elements of “TextLabel” type on the calculation diagram:

1. Select **“Insert”** → **“Primitives panel”** → **“Button”** item in the main menu.
2. Put our such buttons appearing as a table 2x2 onto the diagram (see ).
3. Set names for the buttons: “Bdec1”, “Bdec2” (for the first column), “Binc1”, “Binc2” (for the second column) – acronyms from button decrement, button increment.
4. Select **“Insert”** → **“Primitives panel”** → **“Text”** item in the main menu.
5. Put four such text elements onto the diagram – two ones above the buttons, and two others at the right from the buttons.
6. Write **“-”** and **“+”** in text elements above the buttons, correspondingly.
7. Write texts **“Gstg, t/h =”** and **“Tstg, C=”** in text elements at the right from the buttons, correspondingly (**“Text”** property).
8. Set **“Display the number”** property value to **“Yes”** in the same text elements (at the right from the buttons). Also change **“Displayed value”** property for **“Gstg”** and **“Tstg”**, correspondingly.
9. Increase the font of text notations (using the element property) to 16-20 to mark those out on the calculation diagram.
10. Go to **“Parameters”** tab (at the left from the calculation diagram below **“Diagram”** tab) and set the following four rows there:

if Binc1.Down then Gstg = Gstg + 0.1;

if Bdec1.Down then Gstg = Gstg - 0.1;

if Binc2.Down then Tstg = Tstg + 0.02;

if Bdec2.Down then Tstg = Tstg - 0.02;

These rows will be executed at every step of calculation and, on pressing one or another button, the value of **“Gstg”** variable will be changed by ±0.1 or **“Tstg”** – by ±0.02, which will result in a change in the conditions in the boundary node G and in parameters of steam flow by channels.

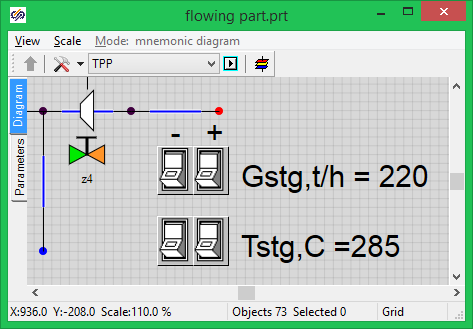


Figure 20. Arrangement of Buttons on the Diagram for Further Debugging of the Model

### Pipelines

To ensure correct operation of the calculation diagram properties of every portion of the pipeline (flowing part) shall be first set, as well as the initial position of all gates. The diagram is still operated independently, without automatics, and, thus, gates can be controlled only manually – either before calculation, on setting the initial position, or in the process of calculation, by manually changing the values of variables storing the positions of the gates.

There are totally nine pipeline portions on the diagram – six alongside of the flowing part, and three others for steam extractions. Set successively the following properties in each common type channel (see an example for the first portion of the flowing part in Figure 21. Channel Parameters at STP Flowing Part Inlet

):

|  |  |
| --- | --- |
| Channel No. 1 | Hydraulic diameter: **“0.3”**  Flow area: **“0.070686”**  Heat transfer surface: **“4.712”**  Length. **“5.0”**  Wall thickness: **“0.01”** |
| Channel No. 2 | Hydraulic diameter: **“0.5317”**  Flow area: **“0.5673”**  Wall thickness: **“0.01”**  Heat transfer surface: **“1.0”** |
| Channel No. 3 | Hydraulic diameter: **“0.5794”**  Flow area: **“1.1805”**  Wall thickness: **“0.01”**  Heat transfer surface: **“1.0”** |
| Channel No. 4 | Hydraulic diameter: **“1.3885”**  Flow area: **“2.0766”**  Wall thickness: **“0.01”**  Heat transfer surface: **“1.0”** |
| Channel No. 5 | Hydraulic diameter: **“1.3885”**  Flow area: **“2.0766”**  Wall thickness: **“0.01”**  Heat transfer surface: **“1.0”** |
| Channel No. 6 | Hydraulic diameter: **“9.9”**  Flow area: **“19.98”**  Wall thickness: **“0.01”**  Heat transfer surface: **“1.0”** |

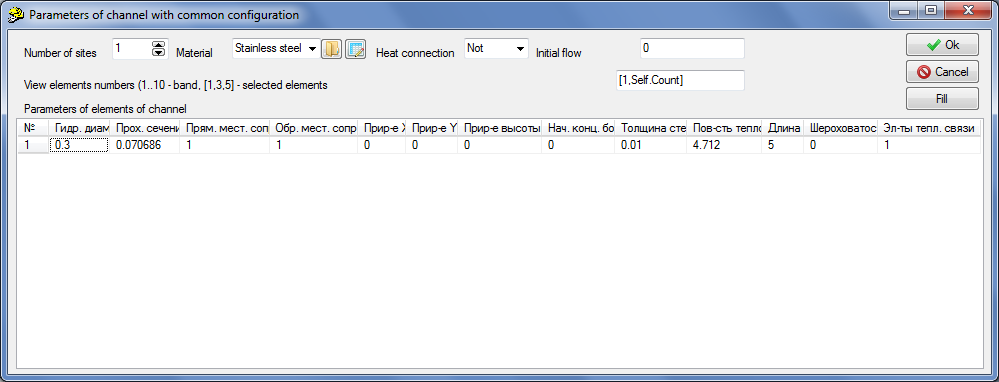


Figure 21. Channel Parameters at STP Flowing Part Inlet

Now it is important for us to set correct hydraulic diameters, flow areas and heat transfer surfaces, the other parameters can still be neglected. Set flowing part piping material as “St20”. It will have effect upon heat transfer.

Set the following properties for extraction pipes:

|  |  |
| --- | --- |
| Extraction channel No. 1 | Hydraulic diameter: **“0.25”**  Flow area: **“0.049087”**  Heat transfer surface: **“3.927”**  Length. **“5.0”**  Wall thickness: **“0.002”** |
| Extraction channel No. 2 | Hydraulic diameter: **“0.5”**  Flow area: **“0.1963”**  Wall thickness: **“0.002”**  Heat transfer surface: **“3.927”** |
| Extraction channel No. 3 | Hydraulic diameter: **“0.35”**  Flow area: **“0.096211”**  Wall thickness: **“0.002”**  Heat transfer surface: **“5.4978”** |

### Gates

Now set the initial position for all gates in percents as equal to 5. File name with “Linear” characteristic, see Figure 22. Synchronous Selection of all Gates on the Diagram

and Figure 23. Synchronous Setting of Properties for all Four Gates

.

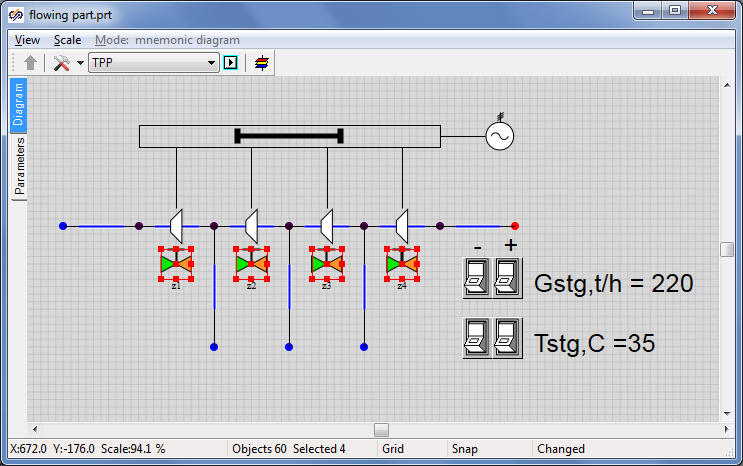


Figure 22. Synchronous Selection of all Gates on the Diagram

In many cases the same values (the same properties) shall be set for several elements of the diagram. In our example the same initial position and the same characteristic shall be set for four gates. Of course, required values can be set for each gate in turn. Nevertheless it is better (quicker) to select all the gates and, on pressing the right-hand button and activating the “Properties” dialog window, to set 5% position and linear characteristic for all the gates at once (see Figure 22. Synchronous Selection of all Gates on the Diagram

and Figure 23. Synchronous Setting of Properties for all Four Gates

). Note that when selecting simultaneously four gates and setting their properties, the “Properties” dialog window display the names of all selected gates.

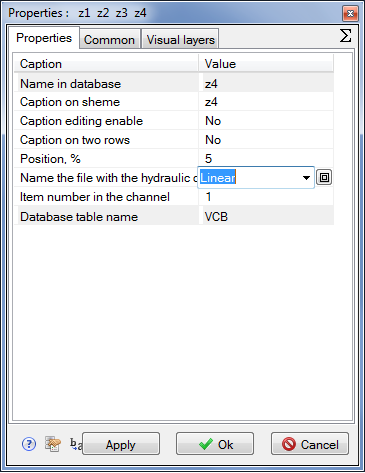


Figure 23. Synchronous Setting of Properties for all Four Gates

### Control of TPP Parameters

Now, provided that you have completed all correctly and without errors, the diagram is ready for normal operation and calculation of the turbine flowing part model. Nevertheless, to view values of particular parameters in the process of calculation, we will need to display these parameters on graphs and/or on the diagram window. Let us use elements of **“TPP parameters control”** tab.

Select “G control in channel” element (control of mass flow in channel), place that on the diagram on the first TPP channel. At the same time make sure that namely this channel is an owner of the newly placed “G control in channel” element (see Figure 24. G Control in the First Channel

).

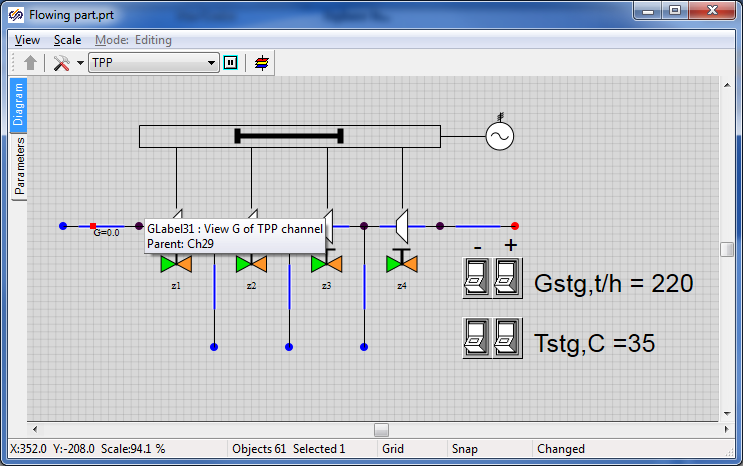


Figure 24. G Control in the First Channel

Go to the element properties and change its two parameters: replace “Text” with “G[t/h]” and “Names of displayed parameters” with “g\*3.6”. Thus we have changed measurement units to the mass flow display. Inside the TPP code the flow will be counted in kilograms per second, while the value will be displayed on the diagram window with coefficient 3.6 s/(kg/t) = 3600 s/ 1000 kg/t (see Figure 25. Conversion to Tons per Hour, “G Control in Channel”

).

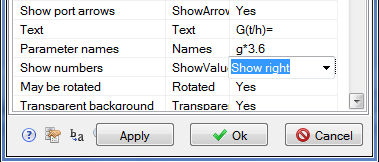


Figure 25. Conversion to Tons per Hour, “G Control in Channel”

Redo the same for all channels; it can be done by copying a newly placed element. Watch correctness of setting of owners of elements being placed. The result shall be as depicted in Figure 26. Arrangement of “G Control in Channel” Elements on Diagram

.

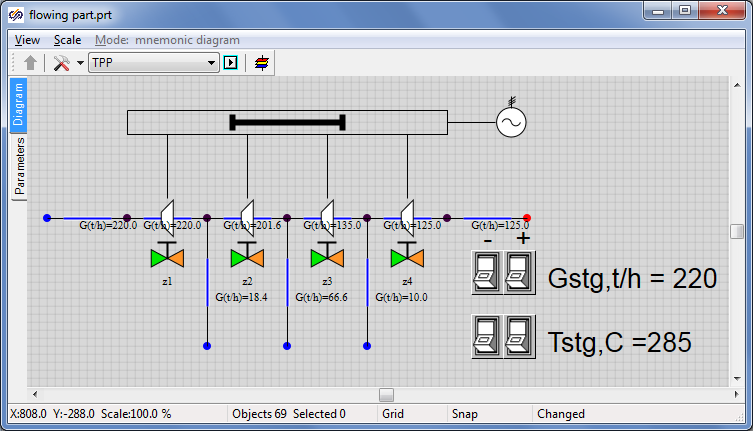


Figure 26. Arrangement of “G Control in Channel” Elements on Diagram

Then, remove pressure and temperature sensors from the diagram in **“P nodes”** (that have been automatically arranged on the diagram on adding the nodes). We will not need those.

We will use “P, H, T control in node” element for all boundary and internal nodes. Arrange 10 such elements on the diagram, see Figure 27. Arrangement of “P, H, T Control in Node” Elements on Diagram

. Pressure, enthalpy and temperature in boundary TPP nodes will be displayed there in the process of calculation. Place the elements with care, watch the owners of these elements. Only one element shall correspond to each point.

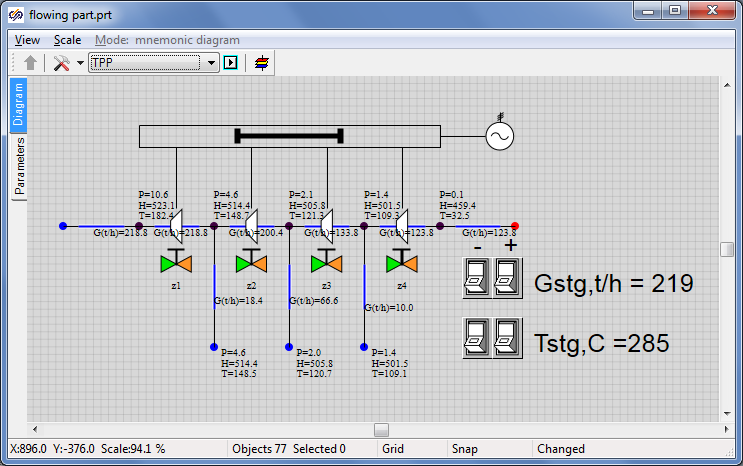


Figure 27. Arrangement of “P, H, T Control in Node” Elements on Diagram

Let us display the rotor rotation speed and generator actual power using the same mechanism but using the **“Parameters”** menu for each element.

Click the mouse left button on the rotor and select “Object parameters” item in a pop-up window. A small dialog window will be displayed, in which a row with “n\_ (Rotation speed)” parameter shall be selected, then press “Create caption” button (with “A” character, see Figure 28. “Rotor” Object Parameters

).

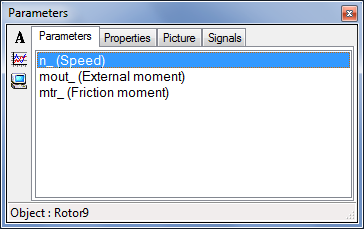


Figure 28. “Rotor” Object Parameters

Leave all unchanged in the next window, as an example the caption can be changed for “Speed =”, see Figure 29. Window for Creation of Animated Caption for Object

:

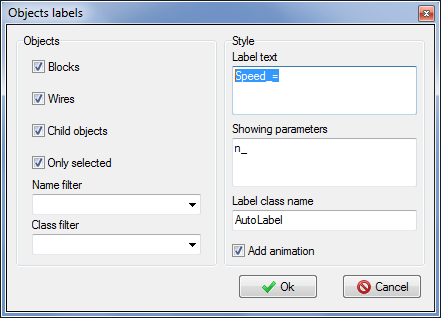


Figure 29. Window for Creation of Animated Caption for Object

As a result we will obtain a new caption on the diagram that will display an actual rotation speed of **“Rotor”** element. Redo the same manipulations and display the electric generator power on the diagram. Go to the properties of the electric generator power caption and change the format of a displayed number for **“Integral”** to avoid any exponent appearing in power displayed on the diagram, see Figure 30. Window for Creation of Animated Caption for “Electric Generator” Object

. Start the diagram for calculation and make sure that newly created caption can work.

Arrange the captions on the diagram as convenient, see Figure 31. Diagram Window (Rotor Speed and Electric Generator Power are Displayed)

.

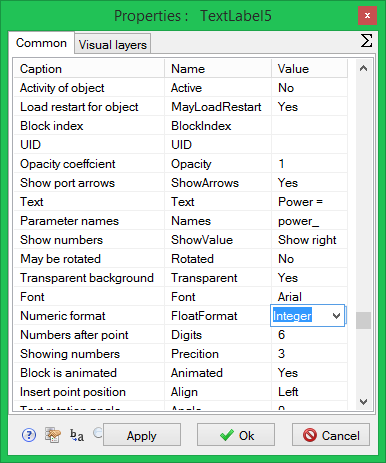


Figure 30. Window for Creation of Animated Caption for “Electric Generator” Object

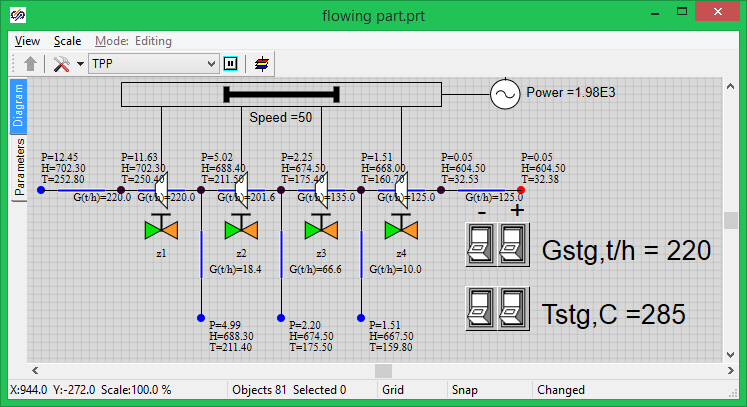


Figure 31. Diagram Window (Rotor Speed and Electric Generator Power are Displayed)

Now the setting of the turbine flowing part is completed, then we will deal with the debugging process for the diagram and rated state of the flowing part considering the initial data.

## Rated State

### General Principles for Debugging the Thermohydraulics Diagram

Prior to starting the work with dynamic modes or connecting regulators and automatics algorithms to the calculation thermohydrulics model, in the general case the diagram shall be debugged so that it ensures a standard and stable calculation and work in rated mode.

To set the rated state we have already taken several important steps, i.e., set parameters for practically all elements (those parameters that we know for sure by their initial data; know that those are required and we are confident about those). In particular, we have set pressure in the condenser and flows in four boundary conditions from the five ones. We have also set steam enthalpies in boundary conditions and geometrical parameters of the channels.

If the diagram is started for calculation we will see that the normal (rated) state has been established in terms of flows: 18.4 t/h, 16.6 and 10.0 t/h is supplied for extraction, 220 t/h of main steam is supplied at the inlet. In this tutorial example we have already set more or less correct values for the channels and have obtained a plausible pressure distribution. In the general case, if we are short of initial data we often have to select hydraulic circuit parameters in order to obtain correct pressures, flows and/or temperatures in reference points of the thermohydraulic path.

If we had not set parameters for the channels (see above) and left the default values, i.e. similar and small flow areas for all channels, then pressure distribution via internal nodes would have been incorrect and practically “arbitrary”.

### Methods for Selecting Flowing Part Parameters

In this tutorial example the turbine flowing part is modeled by equivalent channels. Such a model has right to exist provided that steam parameters in extraction points comply with rated STP parameters. Let us select required drops on the channels and set up the diagram to the rated mode by adjusting pressure drop (loss) on the nodes with gates and moving from the condenser to main steam.

Start the diagram for calculation and set gate **“z4”** in such a position, when pressure in the third extraction point (extraction on PLP No. 1) is equal to 0.96 kg/cm2. To prevent strong jumps and changes of parameters during calculation change the position of the gate gradually with a small increment, e.g., 0.1% or 0.5%. Watch a change of pressure in the third extraction node by setting successively values 5.5, 5.6, 5.7% etc., and restarting the system with every new position of the gate. Proceed with that until a required position of the gate for setting the rated pressure in the node is found. In our case position **“8.1%”** for gate **“z4”** has been obtained.

Then let us go to gate **“z3”** and watch pressure in extraction No. 2. It shall be 3.6 kgf/cm2 when in rated mode. In our example gate **“z3”** shall be set to position “2.35%”.

After that we are selecting the position of gate **“z2”** in the same way. Pressure of 9.2 kgf/cm2 shall be obtained in the first extraction. In our example gate **“z2”** shall be set to position “2.69%”.

Set gate **“z1”** to a position known from the initial data (35 kgf/cm2) for main steam pressure in **“boundary node G”**. In our example gate **“z2”** shall be set to position “1.525%”.

Thus, we have set the rated mode against thermohydraulics parameters of STP flowing part in a first approximation, see Figure 32. Diagram Window with Rated Mode

. Then let us go to power-related (active) elements of the turbine and to setting those to the rated mode.

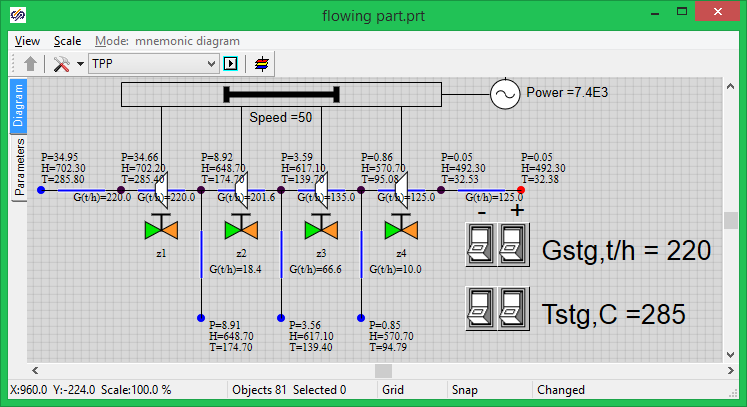


Figure 32. Diagram Window with Rated Mode

### Active Elements, Rotor and Generator

Active elements “pick up” power from hydraulic channels and transmit that to the turbine (generator) rotor.

Set “Mass” flow type for every active element (in properties), and set element characteristic – “tk-35-38-3-st1”, “tk-35-38-3-st2”, “tk-35-38-3-st3”, “tk-35-38-3-st4”, correspondingly, for element 1, 2, 3 and 4, see Figure 33. Properties of Active Element No. 1

. More details about active element characteristic can be found in TPP code documentation. Generally speaking, the characteristic appears as 3 tables, where hydraulic resistance of the channel, efficiency of an element and moment of resistance is set by points depending on the flow (mass flow, in our case) and shaft rotation speed. In our files a constant efficiency is set (equal to 90% or 0.9) for all rotation speeds and flows.

Actual characteristics of equipment can be set in the SimInTech table editor either pinned together in points, or using built-in programming language with the use of cycles, formulas, etc.

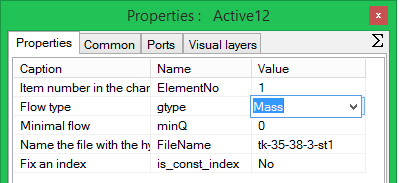


Figure 33. Properties of Active Element No. 1

When properties of active elements are set and the flowing part is debugged for the rated mode, the calculation task can be started and power produced by the electric generator can be seen. In our case the power has been obtained as equal to 7400 kcal ≈ 31000 kW that is in compliance with the initial data.

# 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

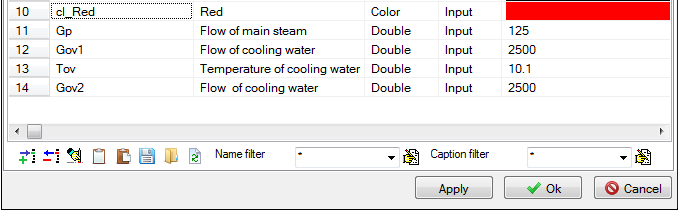


Figure 33. Global parameters for condenser model

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.

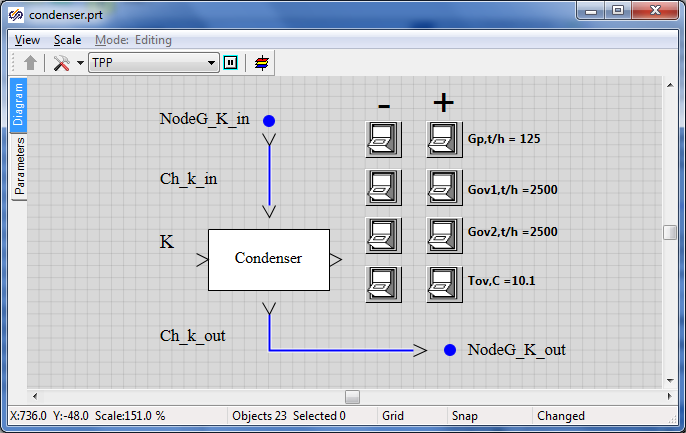


Figure 35. Buttons for changing global parameters in the condenser model

## 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.

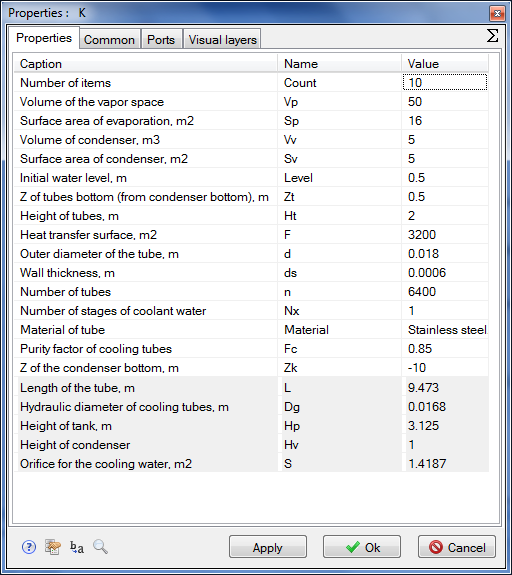


Figure 35. Properties of condenser submodel

### 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:

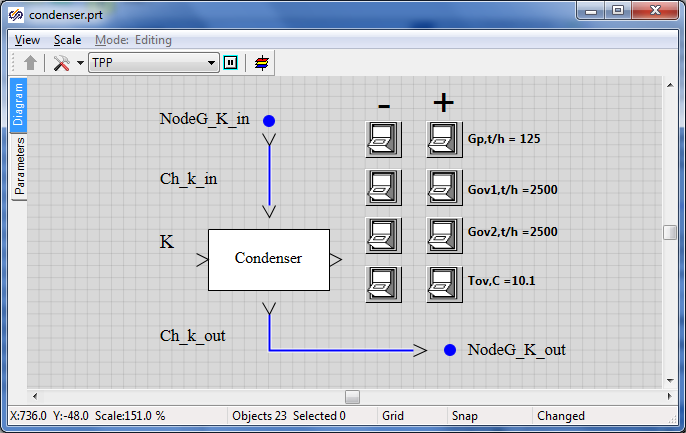


Figure 36. Creation of condenser model (beginning)

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.

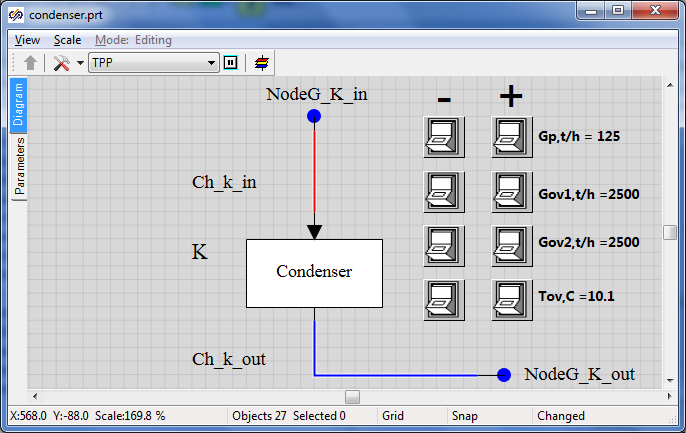


Figure 37. Creation of condenser model (top level is ready)

### 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.

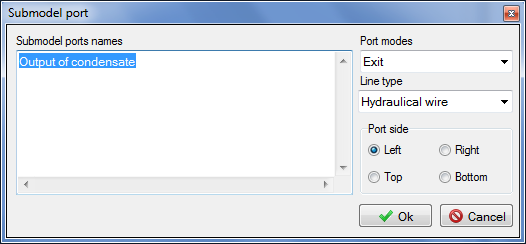


Figure 38. Change of output port name

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.

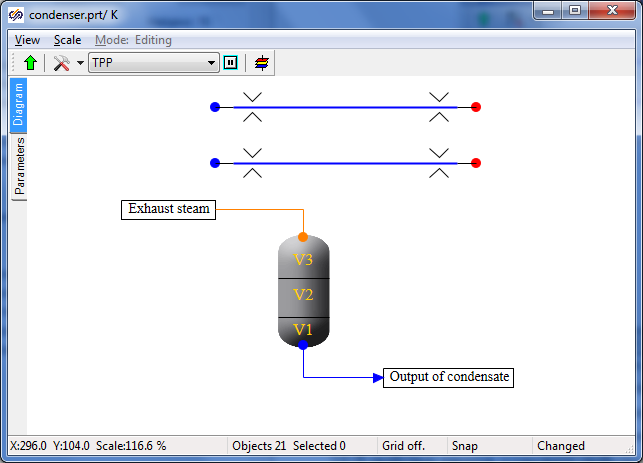


Figure 39. Condenser submodel (nested level)

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.

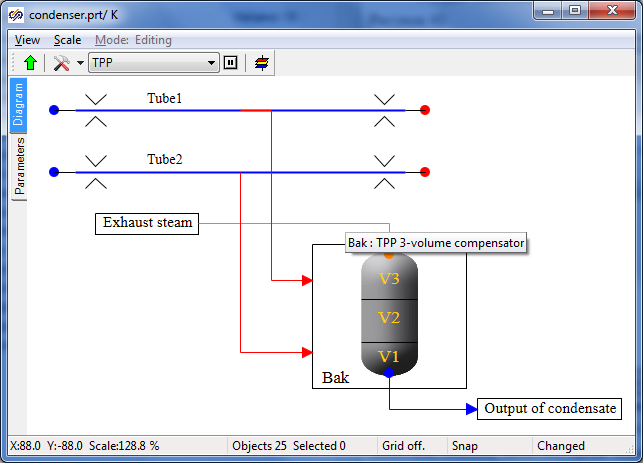


Figure 40. Condenser submodel with heat links (nested level)

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 |

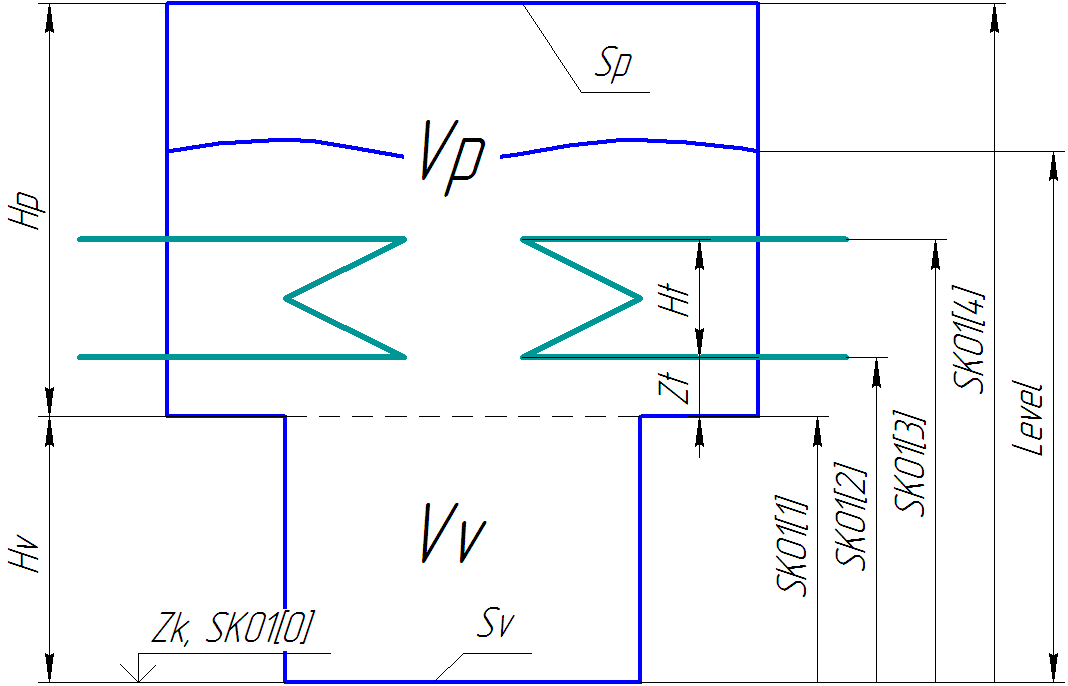


Figure 41. Condenser geometrical model

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).

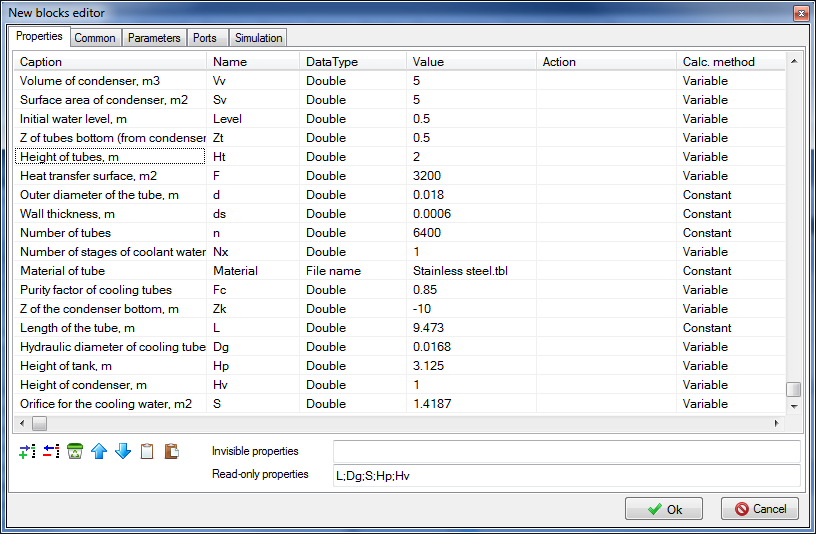


Figure 42. New block editor (condenser)

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.

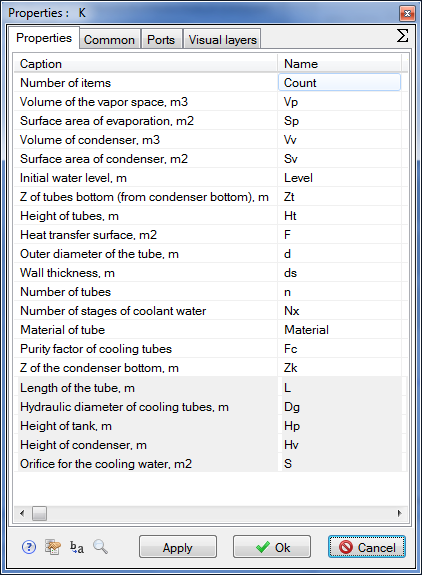


Figure 43. New submodel (condenser) properties

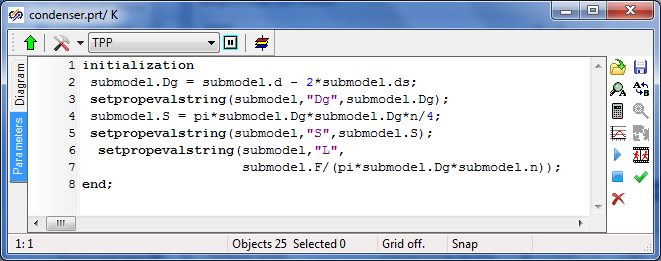


Figure 44. Condenser submodel parameters editor (calculation of properties)

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.

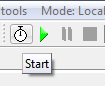


Figure 45. Calculation initialization button

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):

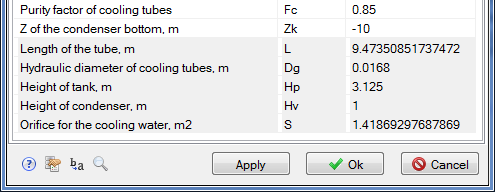


Figure 46. Recalculation of submodel properties

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:

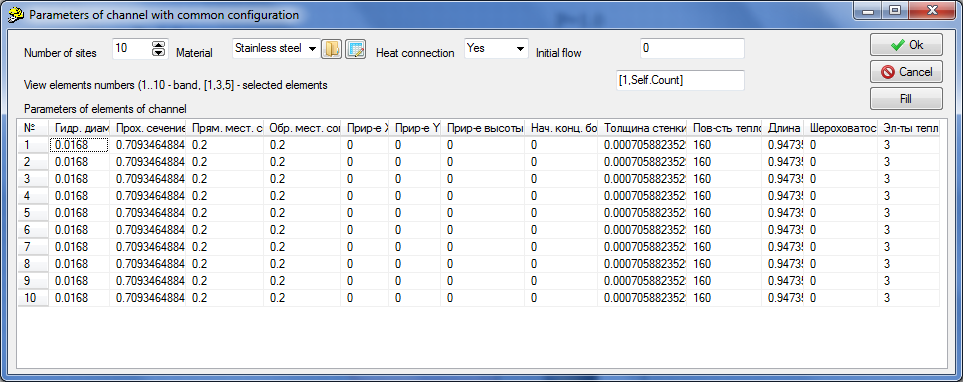


Figure 48. Recalculation of “Tube1” object properties (common-mode channel)

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.

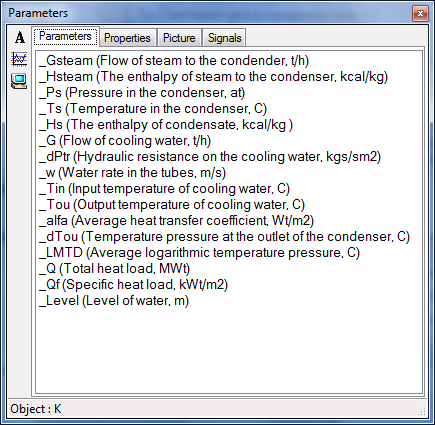


Figure 50. Display of set condenser submodel parameters

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.

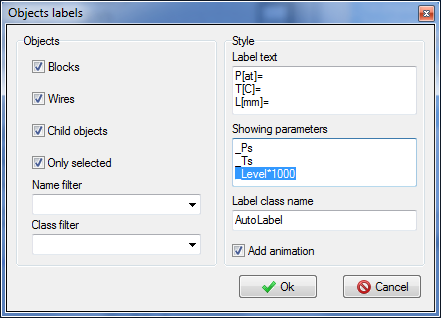


Figure 52. Display of condenser parameters P, T, L in the diagram window

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.

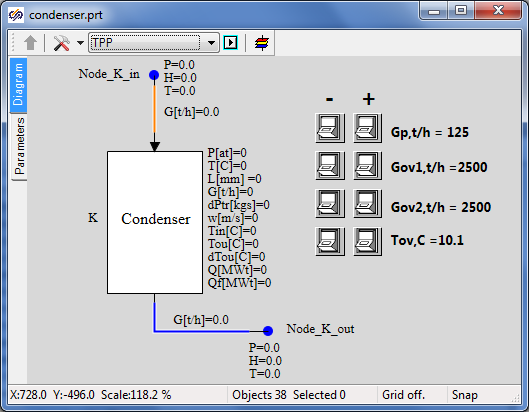


Figure 53. Display of model parameters in diagram window

### 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”**.

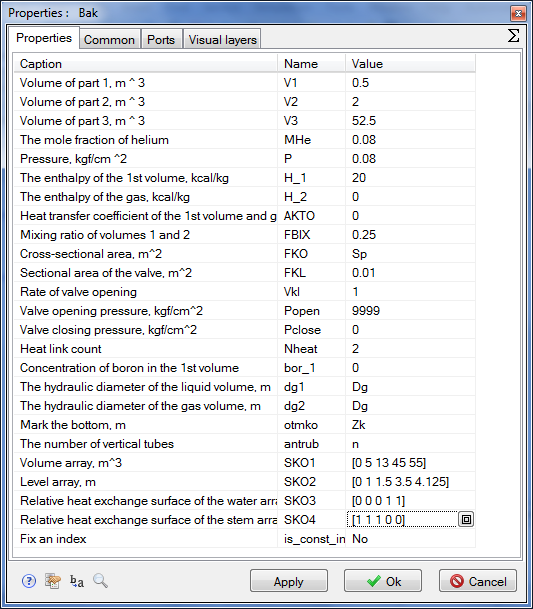


Figure 53. Properties of condenser submodel 3V tank

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.

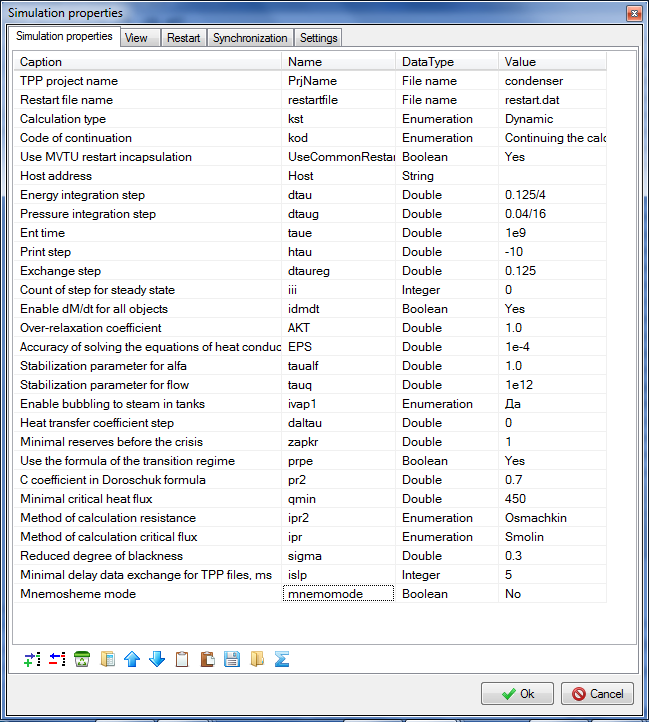


Figure 54. Condenser submodel calculation parameters

### 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.

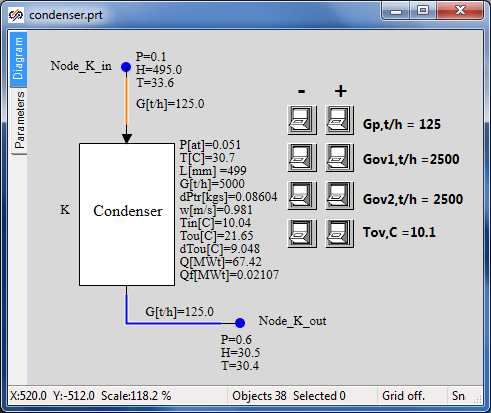


Figure 55. Stationary state of condenser model

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.

# Creation of feed water heater models

We will create thermohydraulics models of three feed water heaters: PND-1 (corresponds to PN‑100), PVD-2 (corresponds to PV-280-1) and PVD-3 (corresponds to PV-280).

## Creation of PND-1 model as a basis for all heaters

### New TPP diagrams

Create three new catalogues: “C:\KTZ\Turbine\PND-1”, “C:\KTZ\Turbine\PVD-2”, “C:\KTZ\Turbine\PVD-3”.

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\PND-1\PN-100.prt”.

Use “**Save as…**” dialog to save the same (empty) project under a new name: “C:\KTZ\Turbine\PVD-2\PV-280-1.prt”.

Use “**Save as…**” dialog to save the same (empty) project under a new name: “C:\KTZ\Turbine\PVD-3\PV-280.prt”.

Figure 56 presents a common diagram for creation (development) of a new thermohydraulics model in SimInTech. Model of condenser in the previous section has been created in accordance with that; models of heaters will be created in the same way. Besides, since models of heaters look like each other then it would be quicker in practice to create one model, completely debug that and, after saving that under another name, introduce minor changes into its structure and/or names of model variables, and, thus, to relatively quickly obtain a ready, about debugged model of new equipment in the end.

Sometimes it happens that a whole system can be taken from previous designs and used as a basis for a new calculation diagram. This also accelerates the process of development of a new model.

For the purpose of training we will create PND-1 model completely (from zero) in **“**C:\KTZ\Turbine\PND-1\PN-100.prt” file, then copy (save) that into “**PV-280-1.prt”** file and, after introducing required minimum changes, debug that and obtain PVD-2 model; then, after saving that model under **“PV-280.prt”** name and introducing changes into boundary conditions and properties of some elements, we will obtain a model of the third heater PVD-3. It can be done faster than creation of a model for every heater every time from zero since the models are different due to names of some variables and values of properties of some blocks. Models PVD-2 and PVD-3 are different only by values of properties and initial conditions.



*Display of parameters in diagram window and on graphs*

*Start-up of the model for calculation, debugging, conditioning to obtain stable nominal state.*

*Setting of parameters (geometrical, hydraulic, thermal ones and others) for all model blocks and elements.*

*Setting model calculation parameters.*

*Creation of new blocks: programming of  
submodel initialization,  
calculation of submodel parameters.*

*Setting of diagram (model structure) in diagram window*

*Setting of model global parameters*

*Creation of a new diagram in a separate file*

Figure 56. General diagram for creation of new hydraulic model in SimInTech

Structure of heater models is the same and is very similar to the condenser – it is a 3V TPP tank with inside heat exchange between flows of steam supplied to the tank from the top and drained from the bottom as condensate and flow of water that is heated when passing through the tank. The condenser is different due to the fact that there is only one heat exchanger and the process of heat exchange is considered from the different point of view and is differently named – steam heats water instead of water cooling steam.

### Setting of model PND-1 global parameters

Open “**C:\KTZ\Turbine\PND-1\PN-100.prt**” file (with a still empty diagram) and set the following global parameters: “**Tplp**” and “**Gplp**”, see Figure 57:

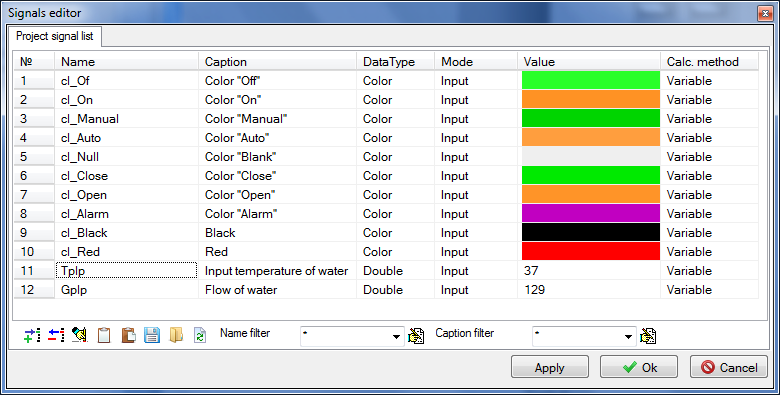


Figure 57. Heater PND-1 model global parameters

Create 4 buttons to control global parameters in the diagram window, see Figure 59. Those will be useful for model debugging. Set the following text in “Parameters” tab to ensure operability of the buttons:

|  |
| --- |
| **if** Binc1.Down **then** Gplp = Gplp+0.1;  **if** Bdec1.Down **then** Gplp = Gplp-0.1;  **if** Binc2.Down **then** Tplp = Tplp+0.02;  **if** Bdec2.Down **then** Tplp = Tplp-0.02; |

### Setting of PND-1 model structure

Open **“**C:\KTZ\Turbine\PND-1\PN-100.prt” file (with a still empty diagram) and set the following elements on the diagram:

1. **“Boundary node G”**, to be used for setting heated water flow.
2. **“Boundary node G”**, to be used for setting flow of steam extraction.
3. **“SimInTech Submodel”**, name of object: **“PN\_100”**, element type (ClassName): **“TPP surface-type heater”** (instead of **“SimInTech submodel”**). Submodel icon is not to be changed, just write **“PN-100”** name for the submodel.
4. **“Boundary node P”** describing parameters of steam in extraction, to be placed above the heater on the diagram.
5. **“Boundary node P”** describing parameters of water supplied to PND-1.
6. **“Common-mode channels”**, 4 elements between each one of boundary nodes and PND-1.

Go inside the submodel and create a model using TPP standard elements:

1. Place 4 elements there – two TPP input ports and two output ports. Rename those as: **“Input of heated water”**, **“Heating steam”**, **“Output of heated water”**, **“Output of steam”**. Change location of ports so that water flows from right to left and steam – from top downward.
2. Place the 3V compenser in the center of the submodel and add one more node (from the top) for steam supply and one thermal port. Rename that as **“Bak”**.
3. Link up input-output ports for water with a common-mode channel and add a heat link to this channel. Rename that as **“Tube”**.
4. Link up the channel and tank with a heat link.
5. Link up the steam inlet to the top tank hole, link up the condensate drain to the bottom tank hole.

The result can be seen in Figure 58.

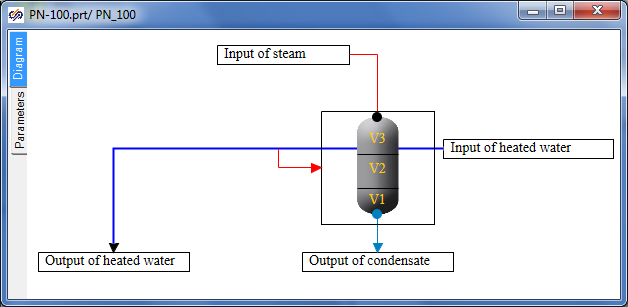


Figure 58. PND-1 heater submodel structure

1. Return to the top level and change the position of appeared submodel ports for a required one (water from left to right, steam from top downward).
2. Link up all elements, compare the diagram obtained with Figure 59.

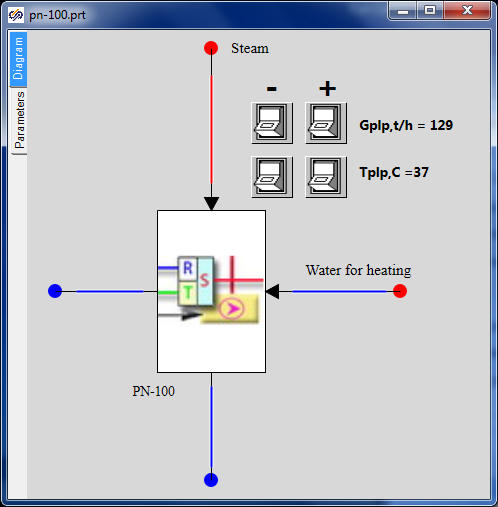


Figure 59. PND-1 heater diagram

### Programming of PND-1 (PN-100) heater submodel

Now, after setting global parameters and the model structure, let us proceed with creation of a new PND-1 block (submodel) and programming of submodel nested level (in a similar way as we have programmed initialization of the turbine condenser in the previous section).

While remaining at the top level of the submodel, select it and use “**Correction**” **→** “**Change block…**” menu item: enter the next 21 properties and 15 parameters for PN-100 submodel, see Figure 60 and Figure 61.

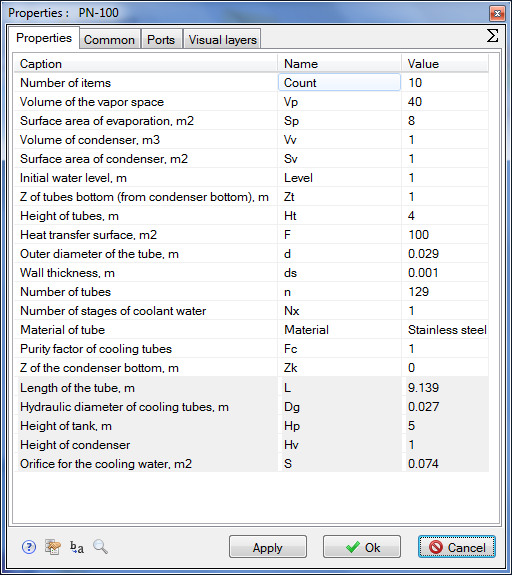


Figure 60. PND-1 heater submodel properties

Do not forget to enter the length, hydraulic diameter of tubes, flow area for cooling water, height of the tank and height of condenser into **“Read-only properties”** field. Enter the parameters and properties carefully and attentively.

After setting the submodel properties and parameters, enter the submodel and go to “Parameters” tab; enter the following code in the initialization block there (try to understand the meaning of every code string by yourself):

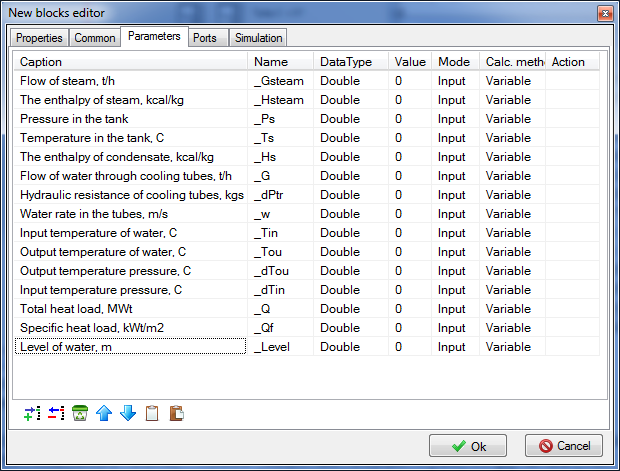


Figure 61. PND-1 heater submodel parameters

|  |
| --- |
| **initialization**  Tube.Material=Material;  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));  **setpropevalstring**(Tube,"Count",submodel.Count);  **setpropevalstring**(Tube,"Gidr\_D","["+Count#submodel.Dg+"]");  **setpropevalstring**(Tube,"Sechen","["+Count#submodel.S+"]");  **setpropevalstring**(Tube,"Dlina","["+Count#(submodel.L/Count)+"]");  **setpropevalstring**(Tube,"Soprot","["+Count#0.0+"]");  **setpropevalstring**(Tube,"InvSopr","["+Count#0.0+"]");  **setpropevalstring**(Tube,"Z","["+Count#0+"]");  **setpropevalstring**(Tube,"X","["+Count#0+"]");  **setpropevalstring**(Tube,"Y","["+Count#0+"]");  **setpropevalstring**(Tube,"Bor","["+Count#0+"]");  **setpropevalstring**(Tube,"Sten","["+Count#(submodel.ds/2/submodel.Fc)+"]");  **setpropevalstring**(Tube,"F","["+Count#(submodel.F/Count)+"]");  **setpropevalstring**(Tube,"Rz1","["+Count#0+"]");  **setpropevalstring**(Tube,"HeatElements","["+Count#3+"]");  InitObject(Tube);    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);  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);  **end;** |

Enter the following strings for calculation of submodel parameters after the initialization block:

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

If everything has been correctly set the submodel has been programmed at that.

### Display of parameters in diagram window

Display pressure and level in the tank, flow for heated water channel at the nested level. Enter P, H, T parameters for nodes, G – for channels and most parameters for the tank, see Figure 62.

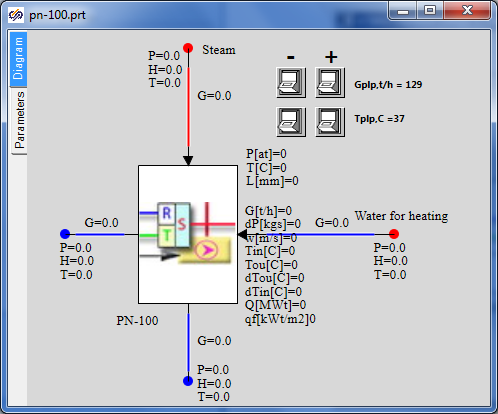


Figure 62. Parameters on PND-1 heater diagram

### Properties of boundary nodes, channels and other elements of PND-1 model

Initialize the diagram in order to check correctness of the entered code and set values for properties of elements inside the submodel (those ones which are set programmatically in the initialization block).

Set the following properties manually for elements of PND-1 model:

|  |  |
| --- | --- |
| Steam supply channel | Hydraulic diameter: **“0.4”**  Flow area: **“0.12567”**  Direct local resistance: **“10”**  Reverse local resistance: **“10”**  Wall Thickness: **“0.005”**  Heat transfer surface: **“6.28319”**  Length: **“5.0”** |
| Condensate drain channel | Hydraulic diameter: **“0.1”**  Flow area: **“0.007854”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.001”**  Heat transfer surface: **“0.31416”**  Length: **“1.0”** |
| Water supply channel (right of the heater) | Hydraulic diameter: **“0.1”**  Flow area: **“0.007854”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.001”**  Heat transfer surface: **“0.31416”**  Length: **“1.0”** |
| Water supply channel (right of the heater) | Hydraulic diameter: **“0.1”**  Flow area: **“0.007854”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.001”**  Heat transfer surface: **“0.31416”**  Length: **“1.0”** |
| Steam extraction node | Pressure: **“0.96”**  Enthalpy: **“573”** |
| Node for water supply for heating | Pressure: **“10.0”**  Enthalpy: **“waterpt(1e6,Tplp,3)/4182”** |
| Heated water extraction node | Flow: **“-Gplp/3.6”**  Enthalpy: **“Tplp”**  Initial pressure: “**10”**  Initial enthalpy: **“self.h”** |
| Condensate extraction node | Flow: **“-ch30.g”** (**“ch30”** – steam supply channel name) |
| Tank | Pressure: **“0.96”**  1st volume enthalpy: **“59.52”**  1st and 2nd volumes dispersion coefficient: **“0.2”**  Section area: **“Sp”**  Valve section area: **“1”**  Fluid volume hydraulic diameter: **“Dg”**  Gas volume hydraulic diameter: **“Dg”**  Z of condenser bottom: **“Zk”**  Number of vertical tubes: **“n”** |
| Top tank node | Initial pressure: **“0.96”**  Initial enthalpy: **“565.8”**  Hydraulic diameter: **“1.4”**  Wall thickness: **“0.014”**  Flow area: **“1.5394”**  Length of node: **“0.5”**  Heat transfer surface: **“2.2”**  Elevation: **“Zk+Hv+Hp”**  Material: **“Ст20”**  No. of volume: **“Steam”** |
| Bottom tank node | Initial pressure: **“0.96”**  Initial enthalpy: **“59.52”**  Hydraulic diameter: **“0.975”**  Wall Thickness: **“0.013”**  Flow area: **“0.7466”**  Length of node: **“0.2”**  Heat transfer surface: **“0.2”**  Elevation: **“Zk”**  Material: **“Ст20”**  No. of volume: **“Lower water”** |

After carefully setting these properties, we have completed creation of PND-1 (PN-100) model. The only thing remained to be done is to set calculation parameters, start up the diagram for calculation and debug the diagram, i.e., to achieve the nominal state of PN-100.

### Parameters of PND-1 calculation

Go to “Calculation parameters” tab and change the following parameters:

|  |  |
| --- | --- |
| Calculation parameters | TPP project name: **“pn\_100”**  Continuation code: **“From the beginning”**  Integration step for energy equations: **«0.125/4»**  Integration step for motion equations: **«0.125/16»** |

Remove the tick from real time synchronization and remove project restart.

### PND-1 nominal state

Now PN-100 model can be started for calculation. If everything has been correctly done, then in 100-300 seconds the state shown in Figure 63 will be set.

Feed water is supplied at +37°С with 129 t/h flow and is heated up to +80°С. At the same time steam (+98°С temperature, 0.96 kgf/cm2 pressure and 11.6 t/h flow) delivers 6.44 MW to the heater. Steam parameters still do not meet nominal values of initial data but at this stage it is important for us to set a correct nominal for feed water line, while steam extractions will be corrected at the stage of integration of individual models into an integral calculation diagram for STP.

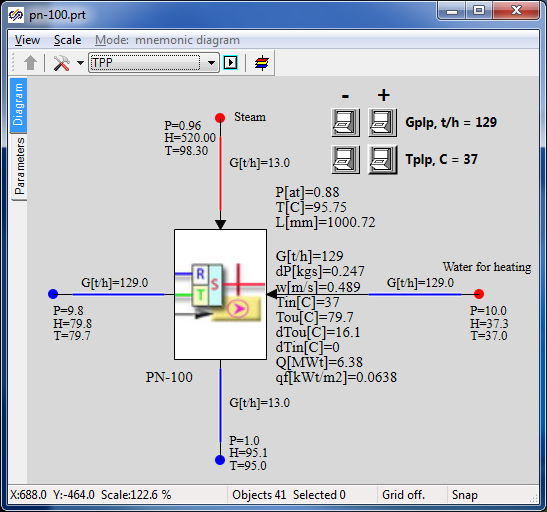


Figure 63. PND-1 nominal state

## Creation of PVD-2 model on the basis of PND-1

### Project copying, calculation parameters

Open the file with PND-1 model created in the previous section and save it into **“**C:\KTZ\Turbine\PVD-2\PV-280-1.prt” file after re-writing the previous file.

Let us begin with renaming the project descriptive parameters: changeTPP project name in the calculation parameters for: **“pv\_280\_1”**, and the submodel name for **“PV\_280\_1”**, and rename the submodel caption as **“PV-280-1”**. Save the project (again).

Thus, we have just created a second heater model in the new file as a copy of the PND-1 model. Further we will proceed with transformation of this model, i.e., we will change only those parts of the model we need to change. Most part will remain the same as in PND-1.

### Global parameters

There will be three global parameters in PVD-2 model: steam pressure in extraction, cooling (heated) water flow and temperature. Create those as per Figure 64. We have changed names of variable a little in order to make sure that the old code will not work with those and, besides, it is more reliable for complete re-checking and changing the model – thus, we will be sure that we have not forgotten to change anything.

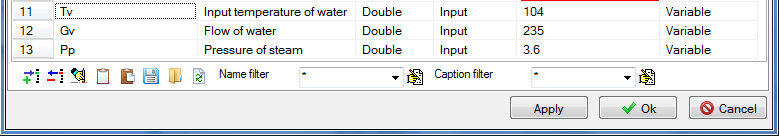


Figure 64. PVD-2 global parameters

When saving these parameters, SimInTech display a warning that Gplp symbol cannot be used. Replace the code in “Parameters” tab with the following strings:

|  |
| --- |
| **if** Binc1.Down **then** Pp = Pp+0.001;  **if** Bdec1.Down **then** Pp = Pp-0.001;  **if** Binc2.Down **then** Gv = Gv+0.1;  **if** Bdec2.Down **then** Gv = Gv-0.1;  **if** Binc3.Down **then** Tv = Tv+0.02;  **if** Bdec3.Down **then** Tv = Tv-0.02; |

Those mean the same but under different names of parameters. In the process of debugging we will change three parameters for PVD-2. In the model of heater we have used the same global parameters in other several places: if you try now to start the diagram for calculation or, at least, to initialize that, SimInTech will display a message about errors in the model since there are still old names **“Gplp”** and **“Tplp”** prescribed in boundary nodes for heated water.

Try to do this and then wherever it is required change the names of parameters for proper ones. I.e., change flow and enthalpy in boundary node G for water and change enthalpy calculation in boundary node P.

### PVD-2 model structure

PVD-2 model structurally does not differ from PND-1 model: the same principle for setting a constant heated water flow with constant parameters at the heater inlet. Steam flow is determined via steam supply and steam parameters set in extraction. Heater heats water and condensates steam along with steam-to-water energy transfer.

PVD-2 model has the same number of boundary nodes, channels and other elements as the model PND-1 has; therefore, we will not structurally change anything.

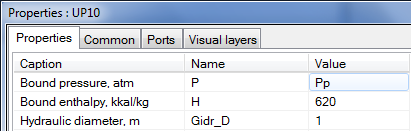


Figure 65. Use of global parameter Pp in PVD-2

The only new thing in the model is another global parameter added, i.e., pressure of extracted steam. Set that, i.e., “**Pp**” instead of **“0.96”**, as a value of parameter for the top boundary node, see Figure 65. Redo the same for the top node in the tank. It could be done also in the properties setting section, but global parameters are a part of the model structure since those can be used to control the model from the very top level.

### PVD-2 submodel

PVD-2 submodel, as well as the structure, has no serious difference from PND-1 submodel and, thus, we will not produce any principle changes here.

As an independent task, change the submodel appearance to make it like depicted in Figure 66 – to that end use **“C:\Program Files\SimInTech\bin\images\Bak-2.gif”** gif-file from the SimInTech delivery set and graphical primitives.



Figure 66. New appearance of PVD-2 submodel

Difference of PND-1 is that some properties of this submodel have other values. Go to **“Change block”** menu item, **“Properties”** tab and change the following parameters:

|  |  |
| --- | --- |
| Submodel properties | Height of tubes, m, “Ht”: **“2”**  Heat transfer surface, m2, “F”: **“280”**  Outer diameter of the tube, m, “d”: **“0.06”** |

An error has been detected in the process of preparation of the training model – in calculation of tubes wall thickness. Correct that and remove division by 2 in the following string:

|  |
| --- |
| **setpropevalstring**(Tube,"Sten","["+Count#(submodel.ds/2/submodel.Fc)+"]"); |

### Display of parameters in diagram window

Since the model structure is the same, all parameters we are interested in have been already displayed in the diagram window, so nothing is to be changed.

### Properties of boundary nodes, channels and other elements of PVD-2 model

Initialize the diagram in order to check correctness of the entered code and reset values for properties of elements inside the submodel (those ones which are set programmatically in the initialization block). Note that the properties marked as “read-only” are also changed since these ones depend on other properties that are also changed.

Now, since steam in PVD-2 is supplied with different parameters and heated water also has different temperature, and diameters of inlet-outlet pipelines are different, change the following properties in the model elements:

|  |  |
| --- | --- |
| Steam supply channel | Hydraulic diameter: **“0.25”**  Flow area: **“0.04909”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.002”**  Heat transfer surface: **“3.927”**  Length. **“5.0”** |
| Condensate outlet channel, water supply channel (right to the heater), water outlet channel (left to the heater) | The parameters will remain the same as for PND-1. |
| Steam extraction node | Pressure: **“Pp” (has been already set)**  Enthalpy: **“620”** |
| Node for water supply for heating | Pressure: **“50.0”**  Enthalpy: **“waterpt(5e6,Tv,3)/4182”** |
| Heated water intake node | Consumption: **“-Gv/3.6”**  Enthalpy: **“Tv”**  Initial pressure: **“10”**  Initial enthalpy: **“self.h”** |
| Condensate intake node | Consumption: **“-ch30.g”** (**“ch30”** – steam supply channel name) |
| Tank | Pressure: **“Pp”**  The other parameters will remain the same as for PND-1. |
| Top tank node | Initial pressure: **“Pp”**  Initial enthalpy: **“620”**  The other parameters will remain the same as for PND-1. |
| Bottom tank node | All parameters may remain the same as for PND-1. |

### Parameters of PVD-2 calculation

We have already changed calculation parameters (project name) in the very beginning, when copying the model. Nothing more is to be changed.

### PVD-2 nominal state

Now, after entering these minimum changes, we can start the diagram for calculation. Nominal state similar to the one depicted in Figure 67 can be set after 200-400 seconds of calculation.

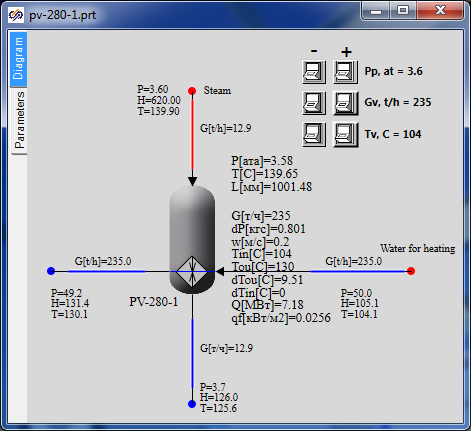


Figure 67. PVD-2 nominal state

Feed water is supplied at +104°С with 235 t/h flow and is heated up to +130°С. At the same time steam (+140°С temperature, 3.6 kgf/cm2 pressure and 13.0 t/h flow) is condensed and delivers 7.18 MW to the heater.

## Creation of PVD-3 model on the basis of PVD-2

### Project copying, calculation parameters

Open the file with PVD-2 model created in the previous section and save it into **“**C:\KTZ\Turbine\PVD-3\PV-280.prt” file after re-writing the previous file.

Let us begin with renaming the project descriptive parameters: changeTPP project name in the calculation parameters for: **“pv\_280”**, and the submodel name for **“PV\_280”**, and rename the submodel caption as **“PV-280”**. Save the project (again).

Thus, we have just created a model of the third heater in a new file as a copy of the model of the second one (PVD-2). Further we will again proceed with transformation of the model, i.e., change of those parts of the model, which shall be changed. Most part will remain the same as in PVD-2.

### Global parameters

There will be three global parameters in PVD-3 model, the same as for PVD-2: steam pressure in extraction, cooling (heated) water flow and temperature. Change their values as per Figure 68. Steam is supplied to PVD-3 under pressure of 9.2 atm and water for heating is supplied with the same flow of 235 t/h and with higher temperature of 130°С.

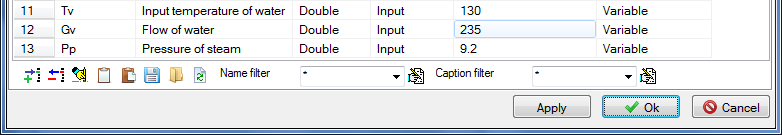


Figure 68. PVD-3 global parameters

Code in “Parameters” tab shall not be changed since the names of global parameters have remained the same. I.e., 6 strings will remain:

|  |
| --- |
| **if** Binc1.Down **then** Pp = Pp+0.001;  **if** Bdec1.Down **then** Pp = Pp-0.001;  **if** Binc2.Down **then** Gv = Gv+0.1;  **if** Bdec2.Down **then** Gv = Gv-0.1;  **if** Binc3.Down **then** Tv = Tv+0.02;  **if** Bdec3.Down **then** Tv = Tv-0.02; |

Since the name of global parameters has not been changed, new values will be automatically assigned for properties in those nodes wherein the parameters are used.

### PVD-3 model structure

PVD-3 model structurally does not differ from PVD-2 model: the same principle for setting a constant heated water flow with constant parameters at the heater inlet. Steam flow is determined via steam supply and steam parameters set in extraction. Heater heats water and condensates steam along with steam-to-water energy transfer.

### PVD-3 submodel

PVD-3 submodel, as well as the structure, has no serious difference from PVD-2 submodel and, thus, we will not produce any principle changes here.

Its difference from PVD-2 is that one property (outer diameter of tubes) of this submodel has another value. Go to **“Change block”** menu item, **“Properties”** tab and change the following parameters:

|  |  |
| --- | --- |
| Submodel properties | Outer diameter of the tube, m, “d”: **“0.044”** |

### Display of parameters in diagram window

Since the model structure is the same, all parameters we are interested in have been already displayed in the diagram window, so nothing is to be changed.

### Properties of boundary nodes, channels and other elements of PVD-2 model

Initialize the diagram in order to check correctness of the entered code and reset values for properties of elements inside the submodel (those ones which are set programmatically in the initialization block).

Now, since steam in PVD-3 is supplied with different parameters and heated water also has different temperature, and diameters of inlet-outlet pipelines are different, change the following properties in the model elements:

|  |  |
| --- | --- |
| Steam inlet channel, condensate outlet channel, water supply channel (right to the heater), water outlet channel (left to the heater) | The parameters will remain the same as for PVD-2. |
| Steam extraction node | Enthalpy: **“650”** |
| Node for water supply for heating | Pressure: **“50.0\*1.02”**  Enthalpy: **“waterpt(50e5,Tv,3)/4182”** |
| Heated water intake node | The parameters will remain the same as for PVD-2. |
| Condensate intake node | The parameters will remain the same as for PVD-2. |
| Tank | The parameters will remain the same as for PVD-2. |
| Top tank node | The parameters will remain the same as for PVD-2. |
| Bottom tank node | The parameters will remain the same as for PVD-2. |

### Parameters of PVD-2 calculation

We have already changed calculation parameters (project name) in the very beginning, when copying the model. Nothing more is to be changed.

### PVD-2 nominal state

Now, after entering these minimum changes, we can start the diagram for calculation: Nominal state similar to the one depicted in Figure 69 can be set after 200-400 seconds of calculation.



Figure 69. PVD-3 nominal state

Feed water is supplied at +130°С with 235 t/h flow and is heated up to +170°С. At the same time steam (+176°С temperature, 9.2 kgf/cm2 pressure and 21 t/h flow) is condensed and delivers 11.1 MW to the heater.

# Creation of models of intermediate circuit heaters

## Creation of model of PS-450 network water heater

### Project copying, calculation parameters

Create “C:\KTZ\Turbine\Network water heater” new catalog.

Open the file with PVD-3 model created in the previous section and save it into **“**C:\KTZ\Turbine\Network water heater\ПС-450.prt” file.

Rename the project descriptive parameters: change TPP project name in the calculation parameters for: **“pv\_450”**, and the submodel name for **“PV\_450”**, and rename the submodel caption as **“ПС‑450”**. Save the project.

Thus, we have just created a network water heater model in the new file as a copy of the PVD-3 model. Further we will proceed with transformation of this model, i.e., we will change only those parts of the model we need to change. Most part will remain the same as in PVD-3.

### Global parameters

There will be three global parameters in PS-450 model: heated water flow and temperature. Change their values as per Figure 70. In PS-450 water for heating is supplied with flow of 420 t/h and temperature of +70°С.

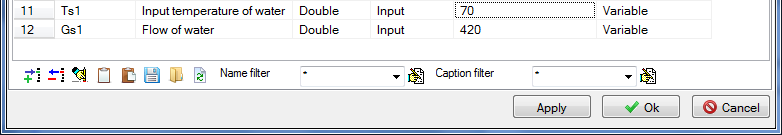


Figure 70. PS-450 global parameters

Code in “Parameters” tab shall be changed since the names of global parameters have been changed (also leave only 4 buttons on the diagram instead of six):

|  |
| --- |
| **if** Binc1.Down **then** Gс1 = Gс1+0.1;  **if** Bdec1.Down **then** Gс1 = Gс1-0.1;  **if** Binc2.Down **then** Tс1 = Tс1+0.02;  **if** Bdec2.Down **then** Tс1 = Tс1-0.02; |

Since the names of global parameters have been changed SimInTech will display an error in those nodes where such names are used. Correct the value of temperature and flow in boundary node G, and set steam pressure to 3.6 kgf/cm2 and enthalpy to 620 in the steam supply node.

### PS-450 model structure

PS-450 model structurally does not differ from PVD-3 model: heated water flow is constant with constant parameters at the heater inlet. Steam flow is determined via steam supply and steam parameters set in extraction. Heater heats water and condensates steam along with steam-to-water energy transfer.

### PS-450 submodel

PS-450 submodel, as well as the structure, has no difference from PVD-3 submodel and, thus, we will not produce any principle changes here.

Its difference from PVD-3 is that one property (outer heat transfer surface) of this submodel has another value. Go to **“Change block”** menu item, **“Properties”** tab and change the following property:

|  |  |
| --- | --- |
| Submodel properties | Heat transfer surface, m2, “F”: **“450”** |

### Display of parameters in diagram window

Since the model structure is the same, all parameters we are interested in have been already displayed in the diagram window, so nothing is to be changed.

### Properties of boundary nodes, channels and other elements of PS-450 model

Initialize the diagram in order to check correctness of the entered code and reset values for properties of elements inside the submodel (those ones which are set programmatically in the initialization block).

Now, since steam in PS-450 is supplied with different parameters and heated water also has different temperature, and diameters of inlet-outlet pipelines are different, change the following properties in the model elements:

|  |  |
| --- | --- |
| Steam supply channel | Hydraulic diameter: **“0.5”**  Flow area: **“0.19635”**  Direct local resistance: “**1”**  Reverse local resistance: **“1”**  Wall Thickness: **“0.005”**  Heat transfer surface: **“7.85398”**  Length. **“5.0”** |
| Condensate outlet channel, water supply channel (right to the heater), water outlet channel (left to the heater) | The parameters will remain the same as for PVD-3. |
| Steam extraction node | Pressure: **“3.6”**  Enthalpy: **“620”** |
| Node for water supply for heating | Pressure: **“25”**  Enthalpy: **“Tc”** |
| Heated water intake node | The parameters will remain the same as for PVD-3. |
| Condensate intake node | The parameters will remain the same as for PVD-3. |
| Tank | Pressure: **“3.6”** |
| Top tank node | The parameters will remain the same as for PVD-3. |
| Bottom tank node | The parameters will remain the same as for PVD-3. |

### PS-450 calculation parameters

We have already changed calculation parameters (project name) in the very beginning, when copying the model. Nothing more needs to be changed.

### PS-450 nominal state

Now, after entering these minimum changes, we can start the diagram for calculation: Nominal state similar to the one depicted in the Figure can be set after 200-400 seconds of calculation.

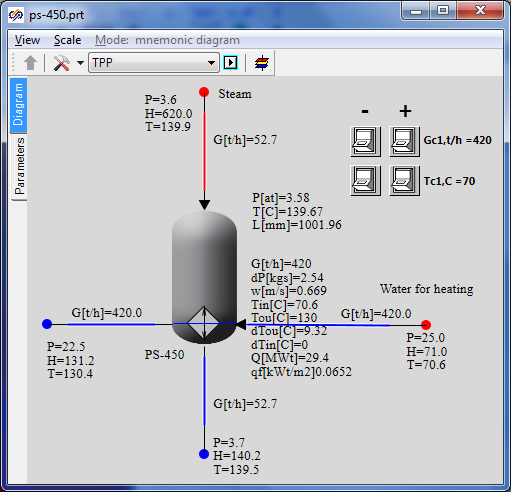


Figure 71. PS-450 nominal state

Network water is supplied at +70°С with 420 t/h flow and is heated up to +130°С. At the same time steam (+140°С temperature, 3.6 kgf/cm2 pressure and 53 t/h flow) is condensed and delivers 29.5 MW to the heater. Steam flow will be adjusted in accordance with rated initial data at the following stages of integration of models into an integrated calculation diagram.

## Creation of model of PS-450P peak heater

### Project copying, calculation parameters

Create “C:\KTZ\Turbine\Peak heater” new catalog.

Open the file with PS-450 model created in the previous section and save it into **“**C:\KTZ\Turbine\Peak heater\PS-450.prt” file.

Rename the project descriptive parameters: change TPP project name in the calculation parameters for: **“pv\_450p”**, and the submodel name for **“PV\_450P”**, and rename the submodel caption as   
**“PS-450P”**. Save the project.

Thus, we have just created a network water heater model in the new file as a copy of the PS-450 model. Further we will proceed with transformation of this model, i.e., we will change only those parts of the model we need to change. Most part will remain the same as in PS-450.

### Global parameters

There will be two global parameters in PS-450P model: heated water flow and temperature. Change their values as per Figure 72. In PS-450P water for heating is supplied with 840 t/h flow and +130°С temperature.



Figure 72. PS-450P global parameters

Change the code in “Parameters” tab since the names of global parameters have been changed.

|  |
| --- |
| **if** Binc1.Down **then** Gpeak = Gpeak+0.1;  **if** Bdec1.Down **then** Gpeak = Gpeak-0.1;  **if** Binc2.Down **then** Tpeak = Tpeak+0.02;  **if** Bdec2.Down **then** Tpeak = Tpeak-0.02; |

Since the names of global parameters have been changed, correct values of temperature and flow in those nodes where such names are used.

### PS-450P model structure

PS-450P model structurally does not differ from PS-450 model. Heated water flow is constant with constant parameters at the heater inlet. Steam flow is determined via steam supply and steam parameters set in extraction. Heater heats water and condensates steam along with steam-to-water energy transfer.

### PS-450P submodel

PS-450P submodel, as well as the structure, has no difference from PS-450 submodel and, thus, we will not produce any principle changes here.

Its difference from PVD-3 is that one property (diameter of tubes) of this submodel has another value. Go to **“Change block”** menu item, **“Properties”** tab and change the following property:

|  |  |
| --- | --- |
| Submodel properties | Outer diameter of the tube, m, “d”: **“0.036”** |

### Display of parameters in diagram window

Since the model structure is the same, all parameters we are interested in have been already displayed in the diagram window, so nothing is to be changed.

### Properties of boundary nodes, channels and other elements of PS-450P model

Initialize the diagram in order to check correctness of the entered code and reset values for properties of elements inside the submodel (those ones which are set programmatically in the initialization block).

Now, since steam in PS-450P is supplied with different parameters and heated water also has different temperature, and diameters of inlet-outlet pipelines are different, change the following properties in the model elements:

|  |  |
| --- | --- |
| Steam supply channel | Hydraulic diameter: **“0.25”**  Flow area: **“0.04909”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.005”**  Heat transfer surface: **“3.927”**  Length: **“5.0”** |
| Condensate outlet channel, water supply channel (at the right), water outlet channel (at the left) | The parameters will remain the same as for PS-450. |
| Steam extraction node | Pressure: **“9.5”**  Enthalpy: **“650”** |
| Node for water supply for heating | Pressure: **“25”**  Enthalpy: **“Tpeak”** |
| Heated water intake node | The parameters will remain the same as for PS-450. |
| Condensate intake node | Enthalpy: **“150”** |
| Tank | Pressure: **“9.5”**  1st volume enthalpy, kkal/kg: **“150”** |
| Top tank node | Initial pressure: **“9.5”**  Initial enthalpy: **“650”**  Hydraulic diameter: **“1”**  Flow area: **“1”**  Heat transfer surface: **“1”** |
| Bottom tank node | The parameters will remain the same as for PS-450. |

### PS-450P calculation parameters

We have already changed calculation parameters (project name) in the very beginning, when copying the model. Nothing more needs to be changed.

### PS-450P nominal state

Now, after entering these minimum changes, we can start the diagram for calculation: Nominal state similar to the one depicted in Figure 73 can be set after 200-400 seconds of calculation.

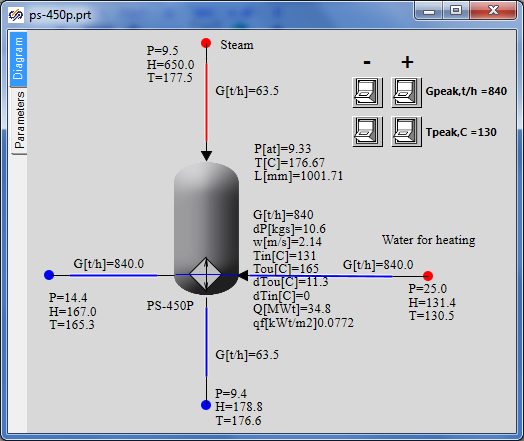


Figure 73. PS-450P nominal state

Network water is supplied at +130°С with 420 t/h flow and is heated up to +165°С. At the same time steam (+170°С temperature, 9.5 kgf/cm2 pressure and 64 t/h flow) is condensed and delivers 35 MW to the heater. Steam flow will be adjusted in accordance with rated initial data at the following stages of integration of models into an integrated calculation diagram.

# Creation of models of pump blocks

## Creation of model of condensate pump block

### New TPP diagram

Create a new TPP project (diagram). Using a standard “save file” dialog save the diagram under a new name in a newly created catalog: “C:\KTZ\Turbine\Condensate pumps\EKN-150-110.prt” (create the catalog in advance).

### EKN-150-110 global parameters

Model of condensate pump blocks is simple, it does not require any global parameters.

### Setting of EKN-150-110 model structure

Structure of condensate pump model is as follows: totally three single-type pumps, common suction for all pumps, upstream of each pump there is a pneumatically driven gate valve, downstream of each pump there is a check valve, a common control valve is fitted on common pressure pipeline.

Place the following elements on the diagram:

1. “**TPP submodel**”, see Figure 74, place all other elements inside that. Rename the submodel as **“MCP”** and caption as **“Condensate pump block”**.

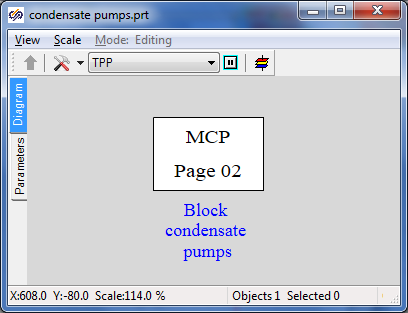


Figure 74. TPP submodel for condensate pump block

1. **“Boundary node P”** to be used for setting pressure in the condenser, i.e., at the pump suction. Place that at the left of the diagram.
2. **“Boundary node P”**, this node is needed just for water outlet, there we will set a random minor pressure (about 10 kgf/cm2) downstream of control valves. Place that at the right of the diagram.
3. “**Internal node TPP**”, place 8 nodes on the diagram – two nodes are common for all pumps and the other six ones shall be placed on three lines for each pump, see Figure 75.
4. **“Common-mode channel”**, 11 elements.

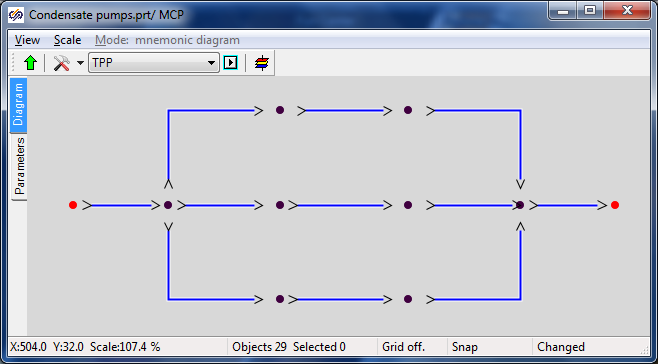


Figure 75. Structure of pipelines and nodes for condensate pumps

1. Link up all channels with the nodes.
2. Place **“Pump without TPP drive”** element (totally three pcs.) on each middle channel.
3. Place **“Gate with TPP pneumatic drive”** elements (totally 4 pcs.) on channels upstream of the pumps and on channel upstream of the right boundary condition.
4. Place **“TPP (typical) check valve”** element (totally three pcs.) on middle channels downstream of the pumps.
5. Compare the result with Figure 76.

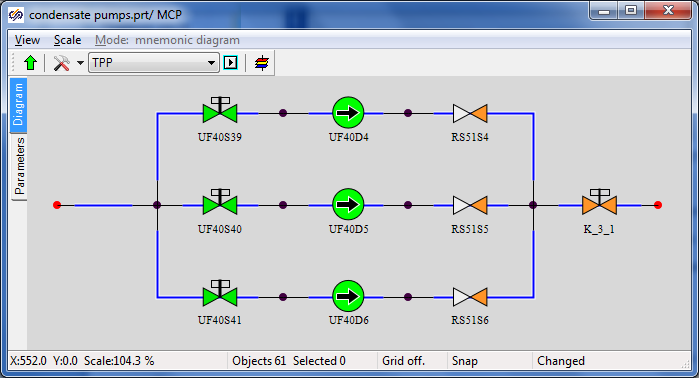


Figure 76. Structure of condensate pump model

1. To test and debug the model we will change the position of the common gate (which is located upstream of the output boundary node). Rename that as **“K\_3\_1”**, set initial position **“100%”** and add two buttons to the diagram to control the gate. Place the following code in “Parameters” tab”:

|  |
| --- |
| **if** Binc.Down **then** K\_3\_1.state = K\_3\_1.state+1;  **if** Bdec.Down **then** K\_3\_1.state = K\_3\_1.state-1; |

1. Rename the pumps as **“MCP-11”**, **“MCP-21”**, **“MCP-31”**.
2. Rename the gates as **“K\_60\_1”**, **“K\_61\_1”**, **“K\_62\_1”**.
3. Rename the check valves as **“K\_63\_1”**, **“K\_64\_1”**, **“K\_65\_1”**.

### Display of parameters in diagram window

1. Display P, H, T parameters (totally 9 pcs.) for all nodes except for the right boundary node.
2. Display the value of flows in t/h (totally 4 pcs.) for all channels downstream of the pumps.
3. Display the current value of control valve in %.

Compare the result with Figure 77.

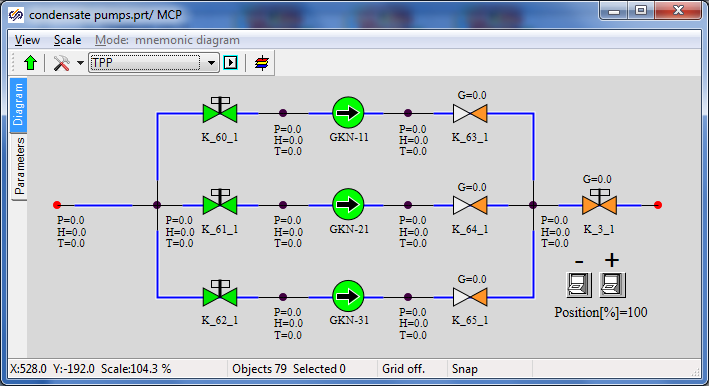


Figure 77. Display of parameters on diagram of model of condensate pumps

### Properties of nodes, channels, pumps and other elements of EKN-150-110 model

For the considered model it is important to correctly set the pump suction pressure, head-capacity characteristics for pumps, correctly set channel diameters and select the line hydraulic resistance with the preset flow.

Set the following properties manually for elements of the condensate pump model (for convenience 3 channels with the same properties can be selected in the beginning to edit their properties together):

|  |  |
| --- | --- |
| Condensate inlet channel | Hydraulic diameter: **“0.5”**  Flow area: **“0.007854”**  Direct local resistance: **“0”**  Reverse local resistance: **“0”**  Wall thickness: **“0.001”**  Heat transfer surface: **“0.31416”**  Length: **“1.0”** |
| Condensate outlet channel (with adjustable gate) | Hydraulic diameter: **“0.15”**  Flow area: **“0.01767”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.002”**  Heat transfer surface: **“2.3562”**  Length: **“5.0”** |
| Water supply channel (3 channels, their properties are the same) | Number of sections: **“2”**  Hydraulic diameter: **“0.2”, “0.2”**  Flow area: **“0.03146”, “0.03146”**  Direct local resistance: **“1”, “1”**  Reverse local resistance: **“1”, “1”**  Wall thickness: **“0.002”, “0.002”**  Heat transfer surface: **“3.1416”, “1.5708”**  Length. **“5.0”, “2.5”** |
| Channel, in which pump is fitted (3 channels, their properties are the same) | Hydraulic diameter: **“0.15”**  Flow area: **“0.01767”**  Direct local resistance: **“25”**  Reverse local resistance: **“25”**  Wall thickness: **“0.002”**  Heat transfer surface: **“1.1781”**  Length: **“2.5”** |
| Pressure channel (3 pcs, downstream of the pump) | Number of sections: **“2”**  Hydraulic diameter: **“0.15”, “0.15”**  Flow area: **“0.01767”, “0.01767”**  Direct local resistance: **“1”, “1”**  Reverse local resistance: **“1”, “1”**  Wall Thickness: **“0.002”, “0.002”**  Heat transfer surface: **“1.1781”, “1.1781”**  Length: **“2.5”, “2.5”** |
| Water supply boundary node (the node describes the condenser) | Pressure: **“0.05”**  Enthalpy: **“32”** |
| Water outlet node (to be left as by-default since water properties are not a matter of interest for us here) | Pressure: **“10”**  Enthalpy: **“30”** |
| Internal nodes (common upstream of pumps, 3 nodes upstream of every pump, 3 nodes downstream of every pump and common node downstream the pumps) | Initial enthalpy: **“30”**  Elevation: **“-20”** |
| Gate “K\_60\_1” | Position: **“100%”** |
| Gate “K\_61\_1” | Position: **“100%”** |
| Gate “K\_62\_1” | Position: **“0%”** |
| Gate “K\_3\_1” | Position: **“100%”** |
| Check valve “K\_63\_1” | Element No. in channel: **“2”**  Pressure drop, at which the channel is open: **“0.01”**  Open channel resistance factor: **“3”**  Closed channel resistance factor: **“1e8”**  Deadband: **“0.001”** |
| Check valve “K\_64\_1” | Element No. in channel: **“2”**  Pressure drop, at which the channel is open: **“0.01”**  Open channel resistance factor: **“3”**  Closed channel resistance factor: **“1e8”**  Deadband: **“0.001”** |
| Check valve “K\_65\_1” | Element No. in channel: **“2”**  Pressure drop, at which the channel is open: **“0.01”**  Open channel resistance factor: **“3”**  Closed channel resistance factor: **“1e8”**  Deadband: **“0.001”** |
| Pump “MCP-11” | Characteristics of pump: **“EKN\_150-110”**  Rotation frequency: **“1”** |
| Pump “MCP-21” | Characteristics of pump: **“EKN\_150-110”**  Rotation frequency: **“1”** |
| Pump “MCP-31” | Characteristics of pump: **“EKN\_150-110”**  Rotation frequency: **“0”** |

Note the setting of pump characteristics – this is a text file name  
“**C:\Program Files\SimInTech\bin\DataBase\Simple pumps\EKN\_150-110.tbl**”. On using the SimInTech tool by selecting “**Tool**” → “**Tables editor**” item menu and opening the file there, one can see (see Figure 78) that the pump head characteristic has been set there: pressure head depending on rotation frequency and volume flow of pumped water. This file has been deliberately created for this model of pumps.

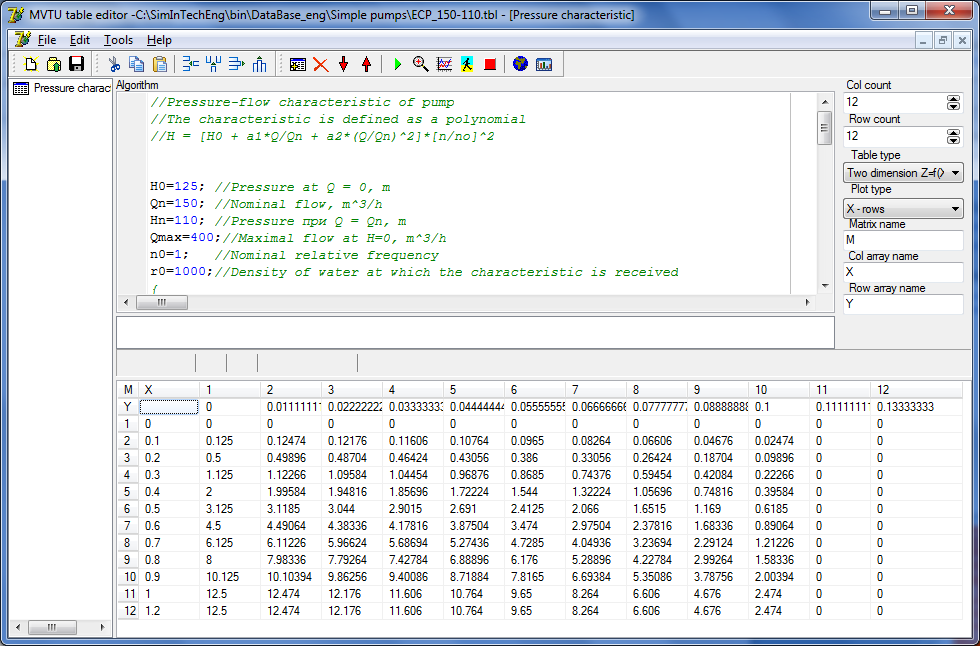


Figure 78. Setting of pump head capacity characteristics

### Nominal state of condensate pump model

On carefully setting these properties, we have completed creation of EKN-150-110 model. Now, if the model is started for calculation, then pressure about 2 kgf/cm2 will be set in nodes upstream of the pumps due to -20 m elevation; two active pumps will be running and create 300 t/h total flow under 12.2 kgf/cm2 pressure at the pump head, see Figure 79. By adjusting the position of the gate at the pump outlet, one can change hydraulic line resistance and debug the pump head capacity characteristics.

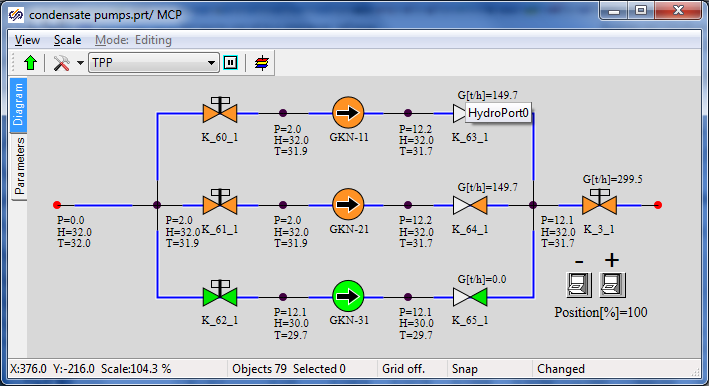


Figure 79. Nominal state of condensate pump block

## Creation of model of feed pump block

### New TPP diagram

Open the project containing the condensate pump model **“**C:\KTZ\Turbine\Condensate pumps\EKN-150-110.prt”. Using a standard “save file” dialog save the diagram under a new name in a newly created catalog: “C:\KTZ\Turbine\Feed pumps\EPN-150-75.prt” (create the catalog in advance).

### EPN-150-75 global parameters

Model of condensate pump blocks is simple; it does not require any global parameters.

### Setting of EPN-150-75 model structure

The structure of feed pump model is very similar to the structure of condensate pump model: totally three single-type pumps, common suction for all pumps, upstream of each pump there is a pneumatically driven gate valve, downstream of each pump there is a check valve and one more pneumatically driven gate (this is a difference from condensate pumps), a common control valve is fitted on common pressure pipeline.

Change the following elements on the diagram:

1. Rename the submodel as “**MFP**” and caption as “**Feed pump block**”, see Figure 80.

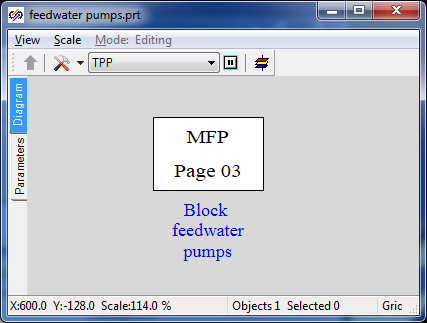


Figure 80. TPP submodel for feed pump block

1. Place additionally three internal nodes inside the submodel.
2. Place additionally three common-mode channels, see Figure 81.
3. Link up all channels with the nodes.

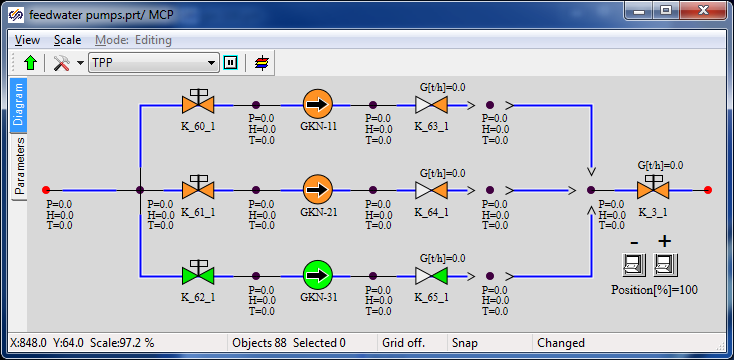


Figure 81. Structure of pipelines and nodes for feed pumps

1. Place **“Pneumatically driven TPP gate”** elements on new channels and name those as **“PV\_14\_1”, “PV\_15\_1”, “PV\_16\_1”**.
2. Place one more pneumatically driven gate **“К\_51\_1”** on the water supply channel.
3. Rename the pumps as **“EPN-11”**, **“EPN-21”**, **“EPN-31”**.
4. Rename the gates as **“K\_53\_1”**, **“K\_54\_1”**, **“K\_55\_1”**.
5. Rename the check valves as **“K\_56\_1”**, **“K\_57\_1”**, **“K\_58\_1”**.

### Display of parameters in diagram window

1. Display P, H, T parameters (totally 3 pcs.) for new nodes.

Compare the result with Figure 82.

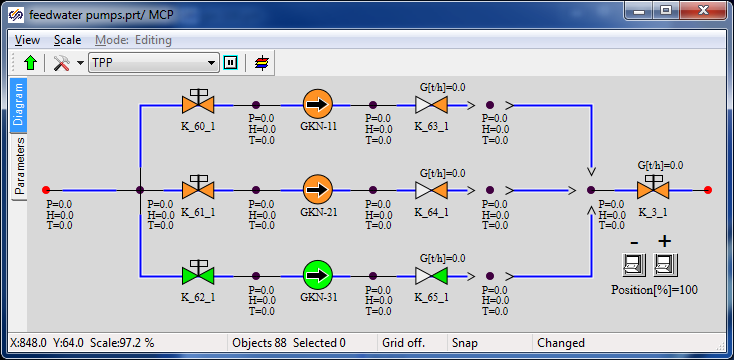


Figure 82. Structure and display of parameters of feed pump model

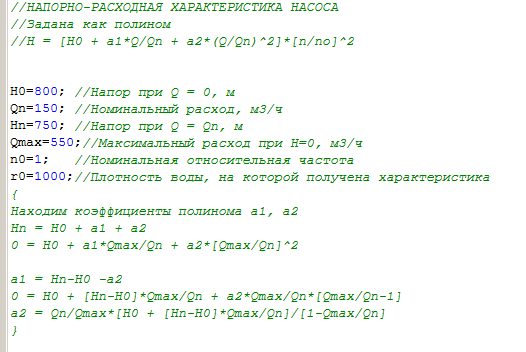
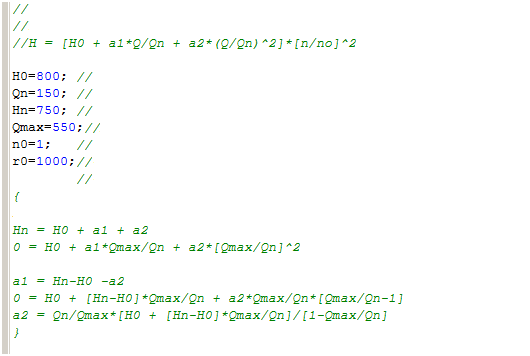
### Properties of nodes, channels, pumps and other elements of EPN-150-75 model

For the model of pumps it is important to correctly set the pump suction pressure, head-capacity characteristics for pumps, correctly set channel diameters and select the line hydraulic resistance with the preset flow.

Set the following properties manually for elements of the feed pump model (for convenience 3 channels with the same properties can be selected in the beginning to edit their properties together):

|  |  |
| --- | --- |
| Water supply channel | Hydraulic diameter: **“0.25”**  Flow area: **“0.04909”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.01”**  Heat transfer surface: **“3.927”**  Length: **“5.0”** |
| Condensate outlet channel (with adjustable gate) | Hydraulic diameter: **“0.15”**  Flow area: **“0.01767”**  Direct local resistance: “**1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.002”**  Heat transfer surface: “**2.3562”**  Length: **“5.0”** |
| Water supply channel (3 channels, their properties are the same) | Hydraulic diameter: **“0.25”**  Flow area: **“0.04909”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.01”**  Heat transfer surface: **“3.927”**  Length: **“5.0”** |
| Channel in which pump is fitted (3 channels, their properties are the same) | Hydraulic diameter: **“0.15”**  Flow area: **“0.01767”**  Direct local resistance: **“25”**  Reverse local resistance: **“25”**  Wall thickness: **“0.005”**  Heat transfer surface: **“2.356”**  Length: **“5”** |
| Pressure channel (6 pcs., downstream of the pumps, with check valves and gates) | Hydraulic diameter: **“0.15”**  Flow area: **“0.01767”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.002”**  Heat transfer surface: **“1.1781”**  Length: **“2.5”** |
| Water supply boundary node (the node describes the deaerator) | Pressure: **“1.2”**  Enthalpy: **“104”** |
| Water outlet node | Pressure: **“10”**  Enthalpy: **“30”** |
| Internal nodes (common upstream of pumps, 3 nodes upstream of every pump, 6 nodes downstream of every pump and common node downstream the pumps) | Initial enthalpy: **“30”**  Elevation: **“0”** |
| Gate “K\_51\_1” | Position: **“100%”** |
| Gate “K\_53\_1” | Position: **“100%”** |
| Gate “K\_54\_1” | Position: **“100%”** |
| Gate “K\_55\_1” | Position: **“0%”** |
| Gate “PV\_14\_1” | Position: **“100%”** |
| Gate “PV\_15\_1” | Position: **“100%”** |
| Gate “PV\_16\_1” | Position: **“0%”** |
| Gate “K\_3\_1” | Position: **“2.987%”** |
| Check valve “K\_63\_1” | Element No. in channel: **“1”** |
| Check valve “K\_64\_1” | Element No. in channel: **“1”** |
| Check valve “K\_65\_1” | Element No. in channel: **“1”** |
| Pump “EPN-11” | Characteristics of pump: **“EPN-150-75”**  Rotation frequency: **“1”** |
| Pump “EPN-21” | Characteristics of pump: **“EPN-150-75”**  Rotation frequency: **“1”** |
| Pump “EPN-31” | Characteristics of pump: **“EPN-150-75”**  Rotation frequency: **“0”** |

Note the setting of pump characteristics – this is a text file name  
“**C:\Program Files\SimInTech\bin\DataBase\Simple pumps\ЭПН\_150-75.tbl**”. On using the SimInTech tool by selecting “**Tool**” → “**Tables editor**” item menu and opening the file there, one can see (see Figure 83) that the pump head characteristic has been set there. Pressure head depending on rotation frequency and volume flow of pumped water. This file has been deliberately created for this model of pumps.



***PUMP HEAD-DISCHARGE CHARACTERISTICS***

***set as polynom***

***Let us find the polynom factors а1, а2***

***Water density, with which the characteristic has been obtained***

***Nominal relative frequency***

***Max. capacity H=0, m3/h***

***Head with Q = Qn, m***

***Nominal capacity, m3/h***

***Head with Q = 0, m***

Figure 83. Fragment of setting of feed pump head capacity characteristic

### Nominal state of condensate pump model

After carefully setting these properties, we have completed creation of EPN-150-75 model. Now, if the model is started for calculation, then due to careful setting of control gate position (2.987%) two active pumps will create 220 t/h total flow, see Figure 84. By adjusting the position of the gate at the pump outlet, one can change hydraulic line resistance and debug the pump head capacity characteristics and the model of feed pump block in general.

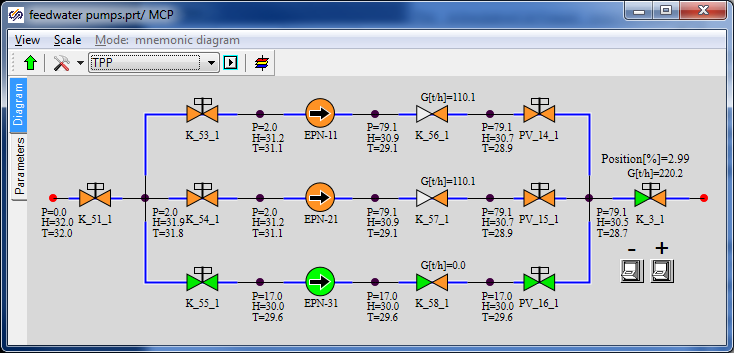


Figure 84. Nominal state of main feed pump block

# Creation of deaerator model

## Description of deaerator model

Aerator model is very simple: this is a 3V TPP tank (compensator) with connected piping for steam supply (from the second extraction) to the deaerator and holes for condensate supply from heaters and hot steam condensate supply from heaters. Water is supplied from the deaerator to the feed pump suction.

## Creation of deaerator model

### Project copying, calculation parameters

Open the file with the feed pump model created in one of the previous section and save it into **“**C:\KTZ\Turbine\Deaerator\Deaerator.prt” file.

Rename the project descriptive parameters: rename TPP project in calculation parameters as “**deair”,** TPP page as **“Deaerator”**, system as **“deair”**, page as **“04”.** Go into the TPP page and, on selecting all that is available inside (except the frame), delete all page content, i.e., remove the feed pump model from there.

Save the project (again). Thus, you have just created a blank for a deaerator model, see  
Figure 85.

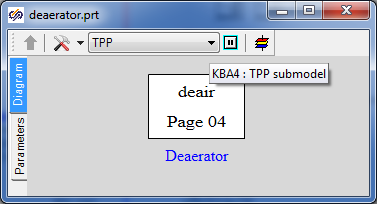


Figure 85. Deaerator model page

### Global parameters of deaerator model

There will be no global parameters in the deaerator model, nothing is to be set.

### Structure of deaerator model

Deaerator model structurally is a 3V TPP compensator with 4 internal holes: two holes in steam zone, one in top water and one in bottom water zone. Piping consisting of three common-mode channels, two internal nodes and one boundary condition is connected to the top hole. Each channel is fitted by one gate – the gate that is closer to the boundary condition is a check valve.

Set such a model yourself according to Figure 86.

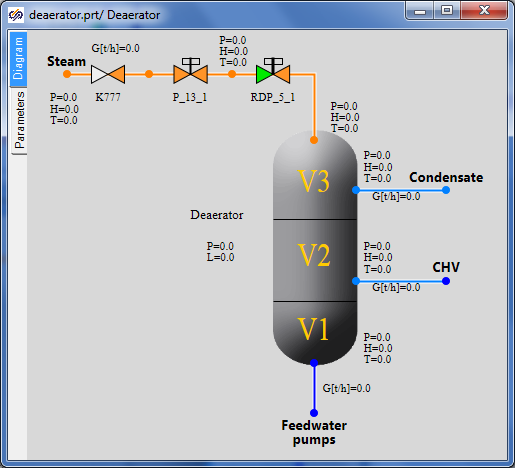


Figure 86. Structure of deaerator model

### Display of parameters in diagram window

Display parameters in the diagram window in accordance with Figure 86.

### Properties of deaerator model elements

At this stage we will accomplish the deaerator model (bring it up to some nominal state) since at the next stage of integration of diagrams into an integral calculation diagram we will connect outputs from other diagrams with parameters that will be there to the deaerator instead of the boundary conditions and channels. Now it is important to set properties of the tank and piping for steam supply to that.

|  |  |
| --- | --- |
| Steam supply channels (3 “common-mode channels” elements) | Hydraulic diameter: **“0.15”**  Flow area: **“0.01767”**  Direct local resistance: **“1”**  Reverse local resistance: **“1”**  Wall thickness: **“0.002”**  Heat transfer surface: **“2.3562”**  Length: **“5.0”** |
| Deaerator tank | 1st part volume: **“1”**  2nd part volume: “**9”**  3rd part volume: **“40”**  Pressure: **“1.2”**  1st volume enthalpy **“104”**  Section area: **“10”**  Valve section area: **“1”**  Valve opening rate: **“0.01”**  Fluid volume hydraulic diameter: **“1”**  Gas volume hydraulic diameter: **“1”**  Number of vertical tubes: “**1”** |
| Top tank node and 2 tank nodes at the middle level (for condensate intake) | Initial pressure: **“7.7”**  Initial enthalpy: **“165.46”**  Hydraulic diameter: **“0.022”**  Wall thickness: **“0.022”**  Flow area: **“0.3848”**  Length of node: **“0.2”**  Heat transfer surface: **“0.44”**  Material: **“Ст20”**  No. of volume: **“Steam”, “Steam”, “Top water”** |
| Bottom tank node | Initial pressure: **“7”**  Initial enthalpy: **“165.8”**  Hydraulic diameter: **“0.7”**  Wall Thickness: **“0.02”**  Flow area: **“1.53938”**  Length of node: **“0.1”**  Heat transfer surface: **“0.2198”**  Elevation: **“-12.7”**  Material: **“Ст20”**  No. of volume: **“Lower water”** |
| Valve K777 | Element No. in channel: **“1”**  Pressure drop, at which the channel is open: **“0.01”**  Open channel resistance factor: **“3”**  Closed channel resistance factor: **“1e8”**  Deadband: **“0.001”** |
| Gate P\_13\_1 | Position: **“100%”** |
| Gate RDP\_5\_1 | Position: **“2%”** |

### Parameters of deaerator calculation

We have already changed calculation parameters (project name, etc.) in the very beginning, when copying the model. Nothing more is to be changed.

### Nominal state of deaerator

On creating the deaerator model, we could have got a nominal state if we had correctly set boundary conditions. Nevertheless, since it is hard to say what nominal water flows and properties will be at deaerator inlets, then we have to be limited with creation of a model – its debugging will be done later at the process of integration of models, that is, in the process of creation of a full thermohydraiulic diagram of STP.

When writing the technique and starting the deaerator model, the lower boundary condition was replaced with a node of G-type with “-220/3.6” kg/s flow, while properties of boundary conditions of P‑type were selected for one of the states of the deaerator; the following result was obtained: see Figure 87.

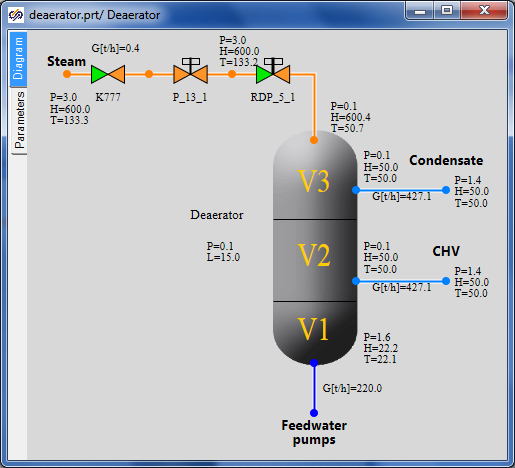
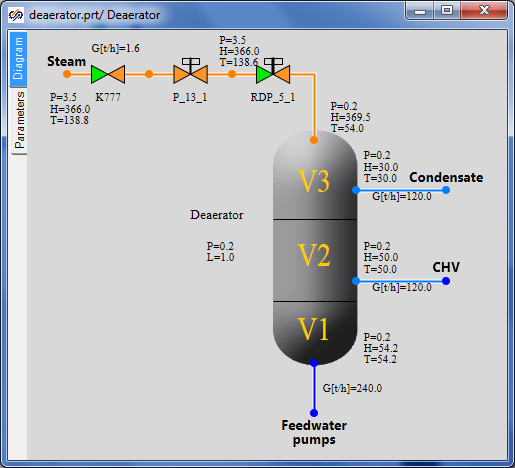


Figure 87. State of deaerator model

# 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”**.

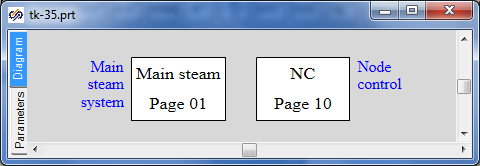


Figure 88. Main Steam System, Beginning of STP Integrated Model Generation

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.

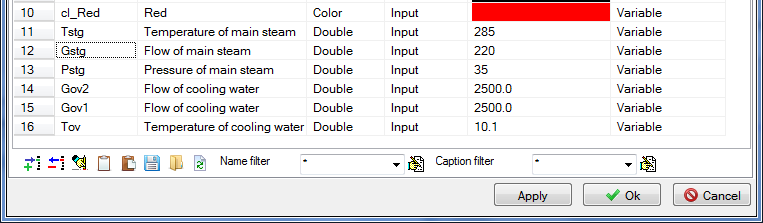


Figure 89. Main Steam System, Six Global Signals

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.

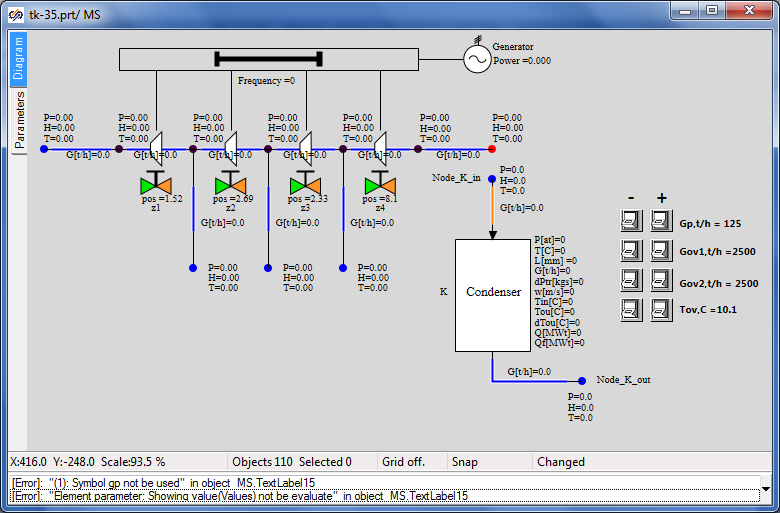


Figure 90. Integration of the Flow Part and Condenser Models

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.

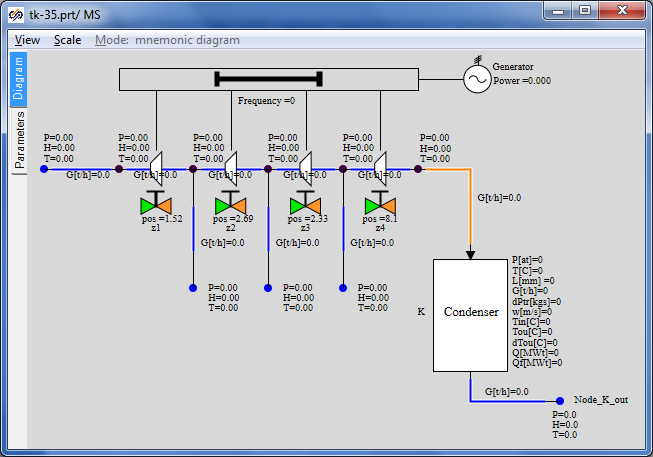


Figure 91. Structure of Main Steam System (Flowing Part + Condenser)

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.

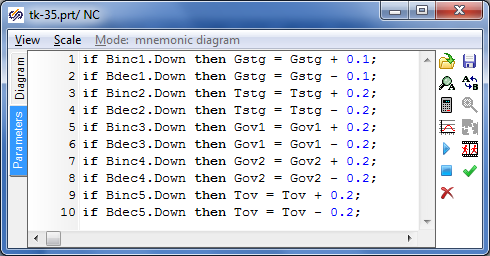


Figure 92. Button Control of Main Steam System Boundary Conditions

### 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.

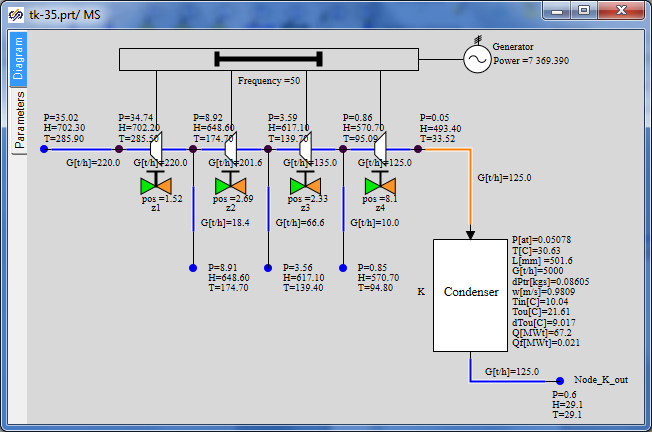


Figure 93. Nominal State of Main Steam System

### 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”** |

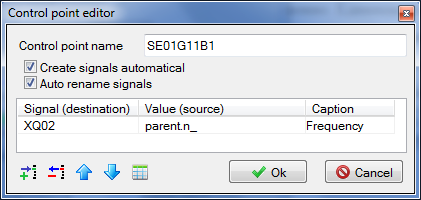


Figure 94. Rotor Rotation Speed Control Point

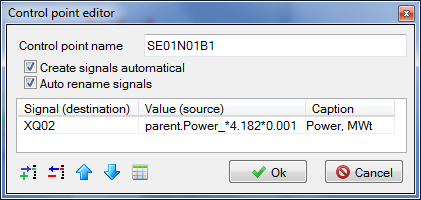


Figure 95. Generator Power Control Point

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.

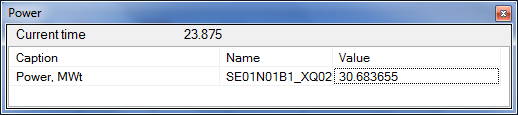


Figure 97. “Generator Power” Parameter Display Window

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.

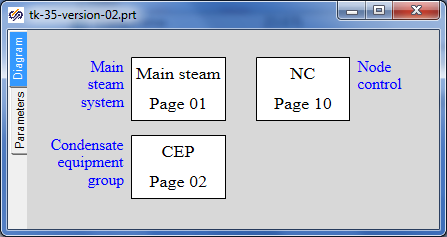


Figure 98. New TPP Page for Condensate Equipment Group

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.

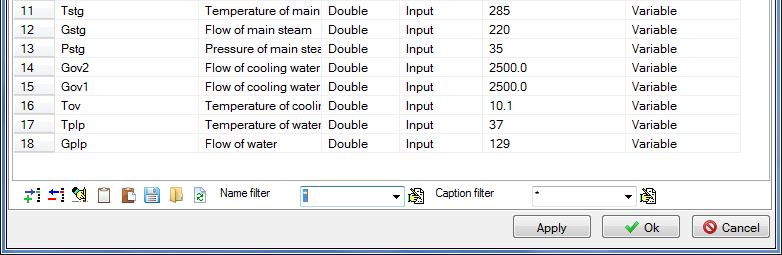


Figure 99. Additional of Gplp and Tplp signals

### 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.

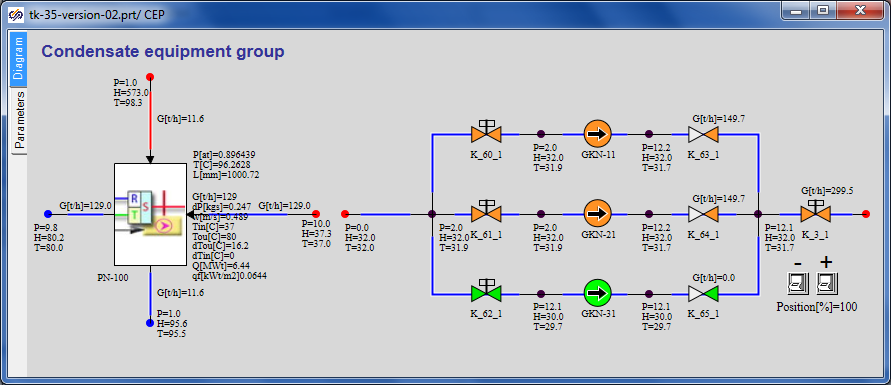


Figure 100. Generation of Condensate Equipment Group

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.

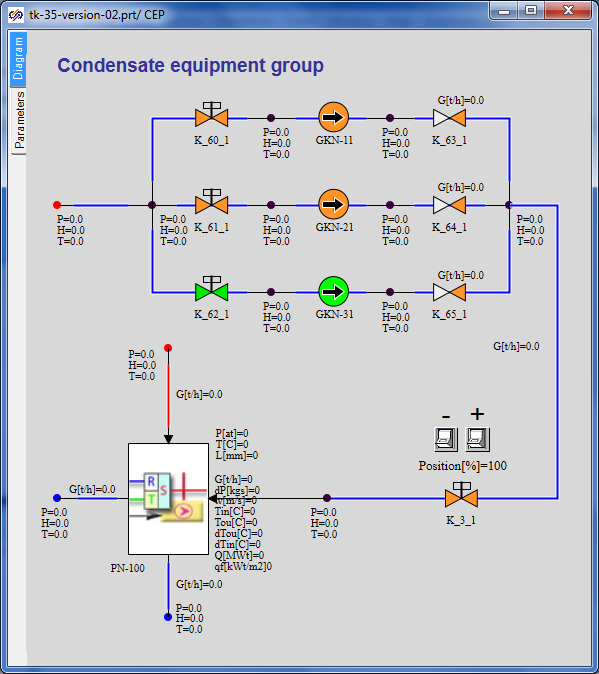


Figure 101. Connection of PLP-1 and Condensate Pumps

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.

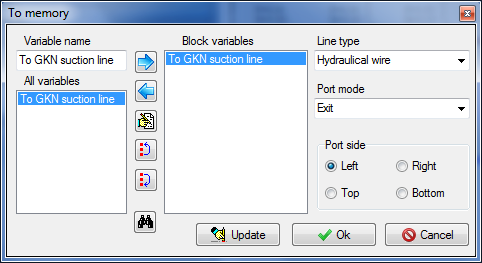


Figure 102. Creation of a New Variable in TPP Memory

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.

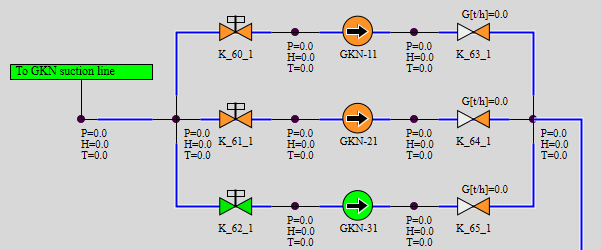


Figure 103. Connection of Suction Line of Condensate Pumps with Condenser

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).

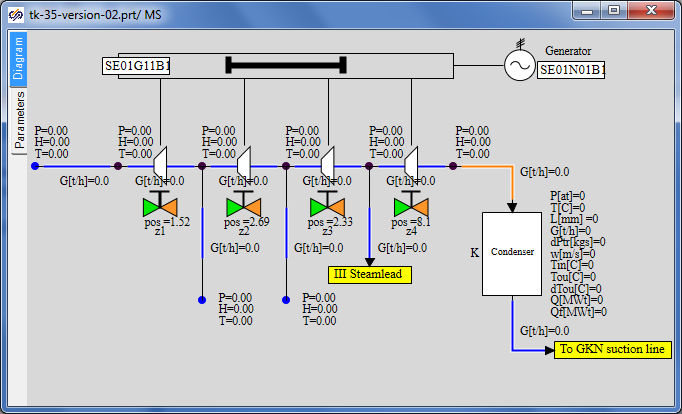


Figure 104. Connection of the Third Extraction and PLP-1

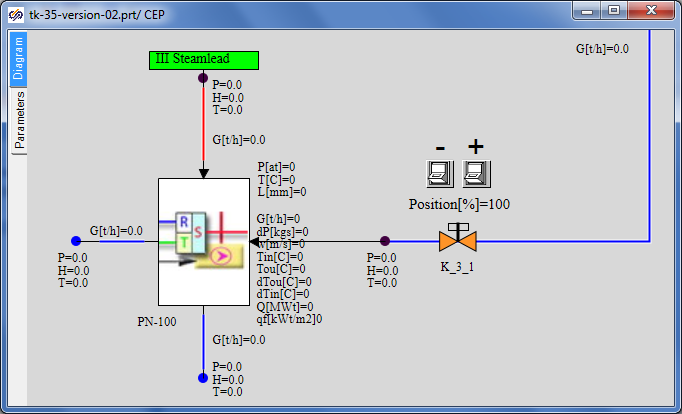


Figure 105. Connection of PLP-1 and the Third Extraction

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.

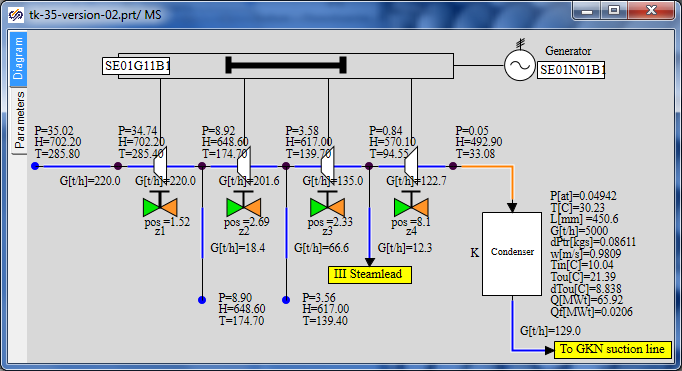


Figure 106. Stationary Calculation of Connection of PLP-1 and the Third Extraction

## 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”.**

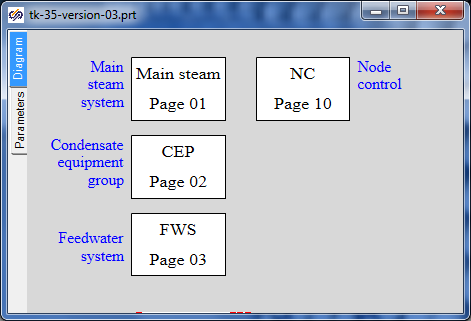


Figure 107. Feedwater System Submodel

### 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.

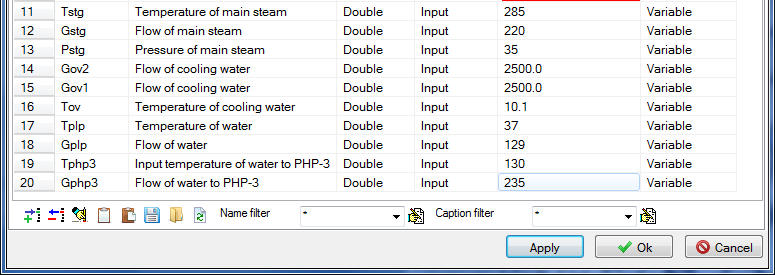


Figure 108. Addition of Signals Gphp3 and Tphp3

### 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.

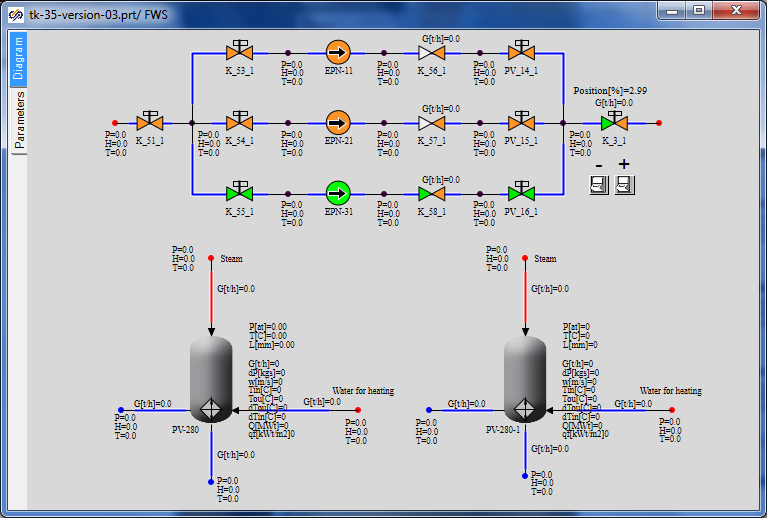


Figure 109. Generation of Feedwater System Submodel

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.

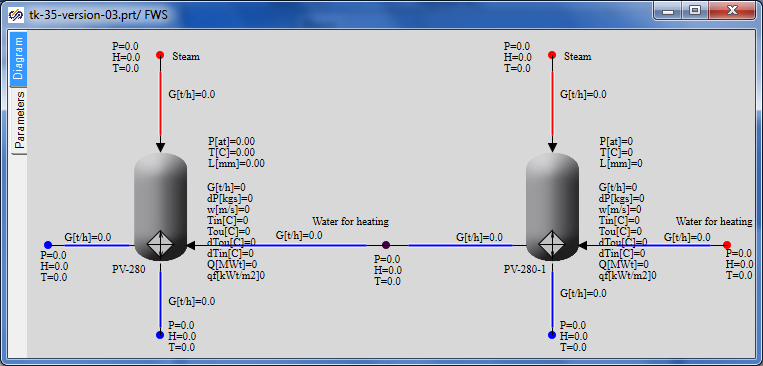


Figure 110. Connection of PHP-2 and PHP-3 via Water

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.

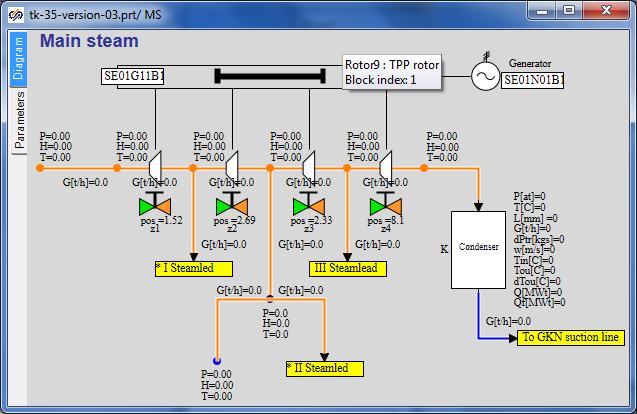


Figure 111. Arrangement of Steam Extraction I and II

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.

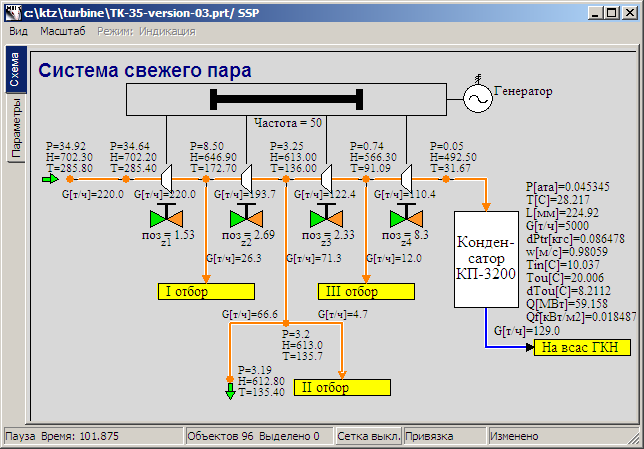


Figure 112. State of Main Steam System after Adding Steam Extractions

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.

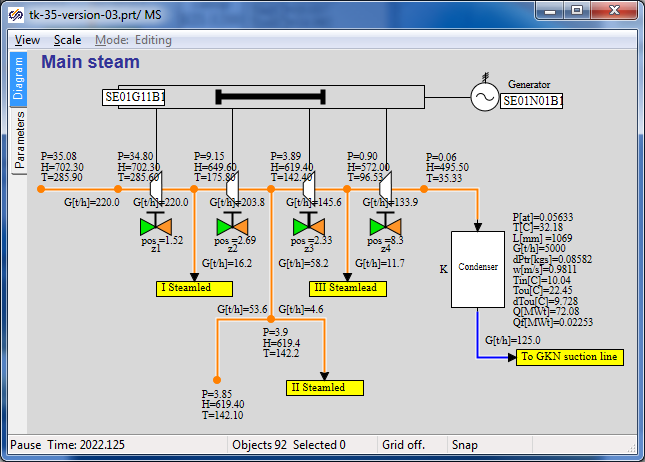


Figure 113. State of Main Steam System after Rectification of Gross Errors

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.

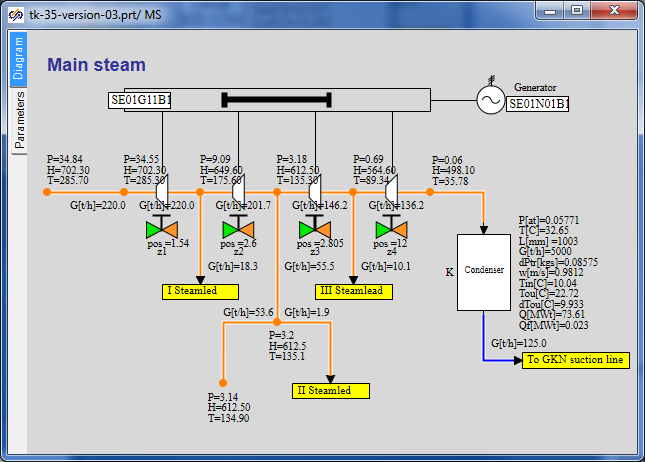


Figure 114. State of Main Steam System after Change of Positions of Gates

## 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.

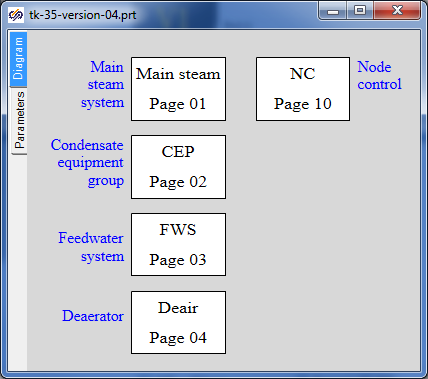


Figure 115. Addition of Deaerator Submodel Page

### 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.

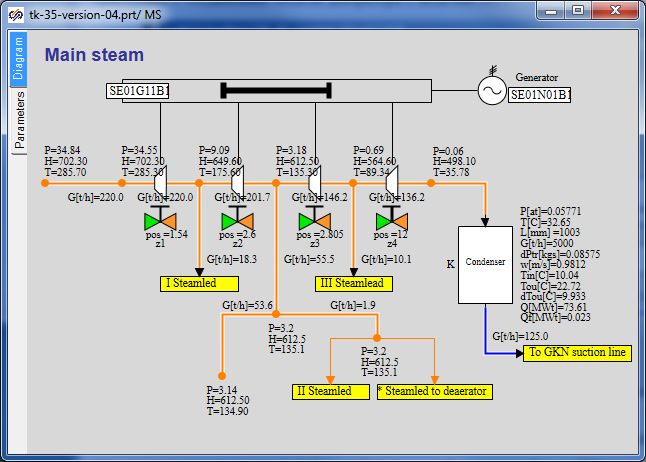


Figure 116. Addition of New Extraction to Deaerator

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.

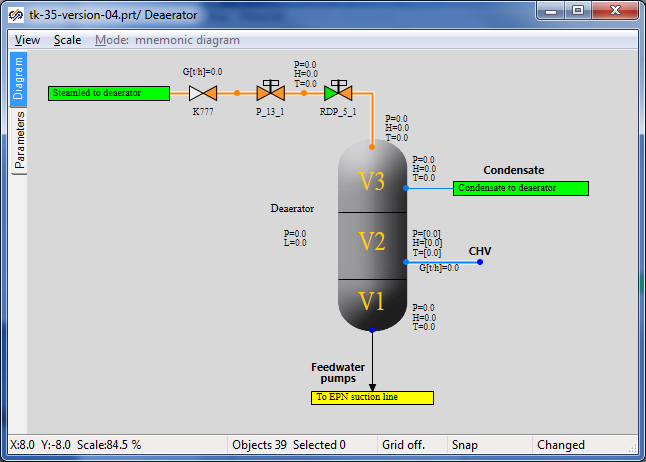


Figure 117. Deaerator Submodel with Links Connected

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.

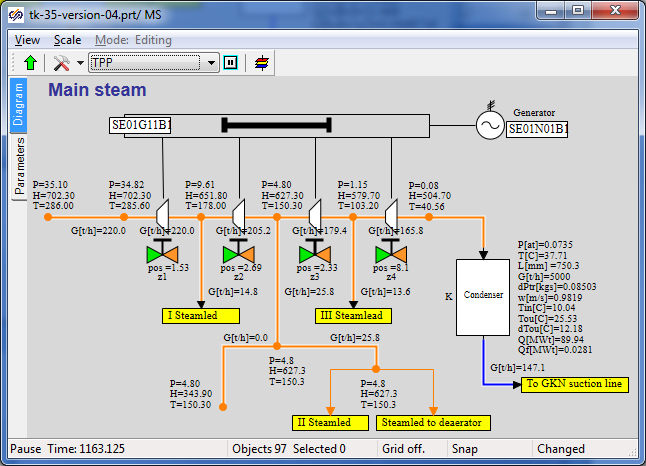


Figure 118. Main Steam System Submodel, Stationary State

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.

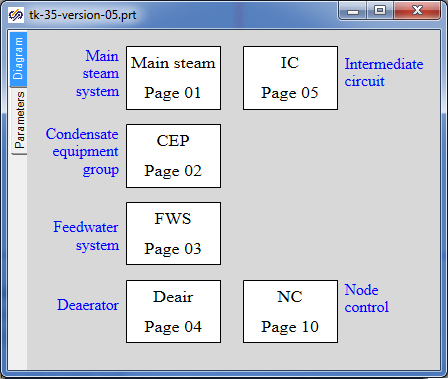


Figure 119. Addition of Intermediate Circuit Submodel Page

### 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.

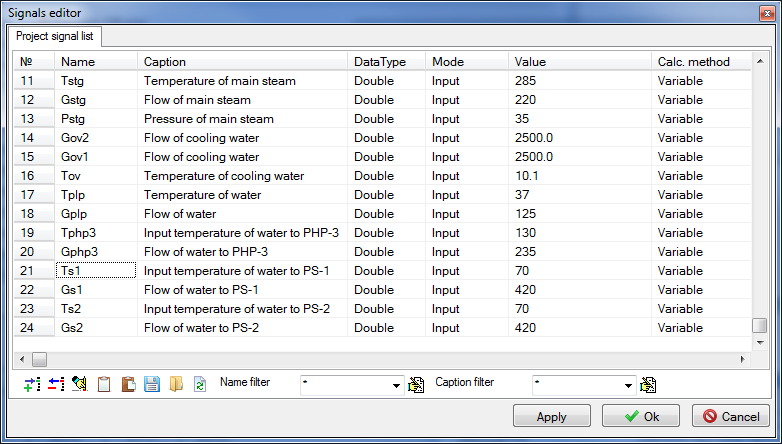


Figure 120. Global Project Signals

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.

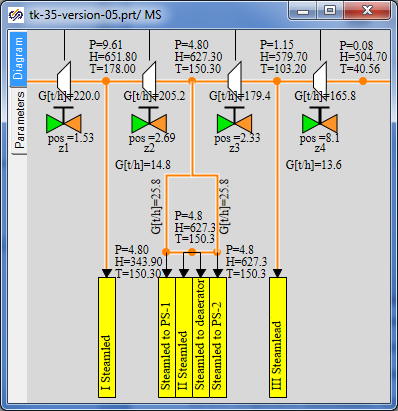


Figure 121. Extractions to Intermediate Circuit Heaters

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.

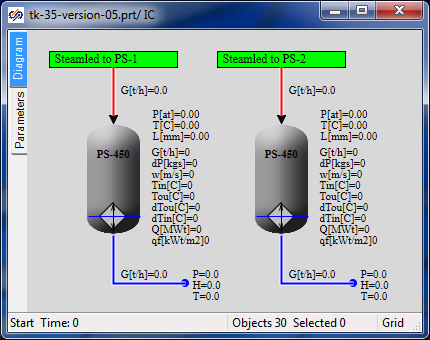


Figure 122. Steam Supply to Intermediate Circuit Heaters

### 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.

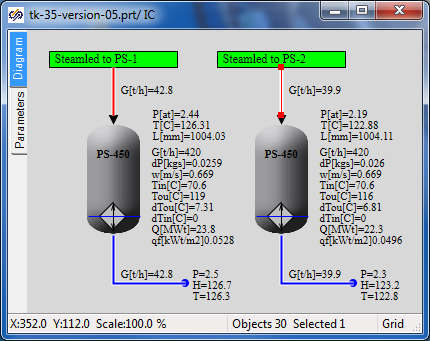


Figure 123. Stationary State of Intermediate Circuit Heaters

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.

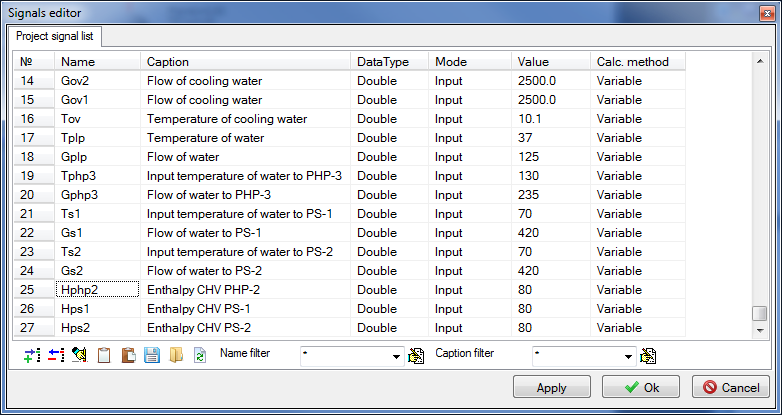


Figure 124. Connection of PHP-2, PS-1 and PS-2 with Deaerator, Global Signals

### 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.

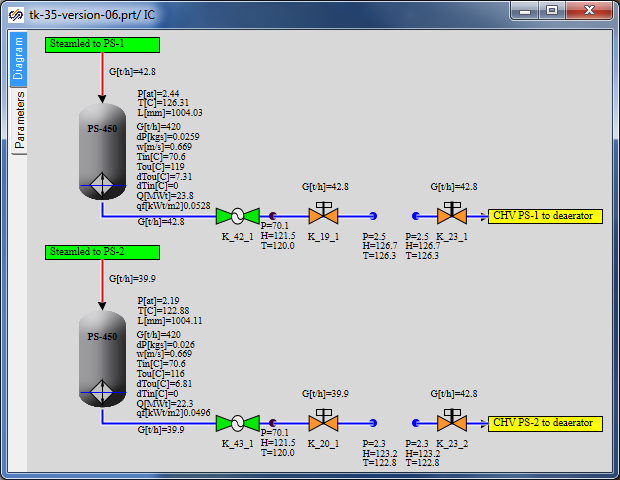


Figure 125. Connection of PS-1 and PS-2 to Deaerator, Addition of Level Controllers

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.

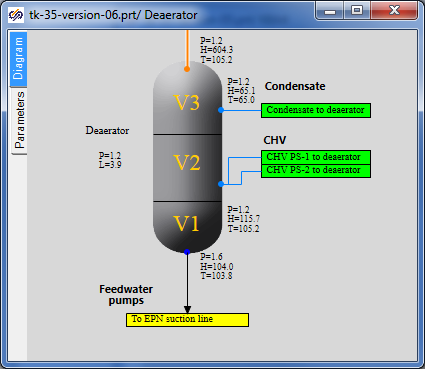


Figure 126. Connection of Intermediate Circuit Heater to Deaerator

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).

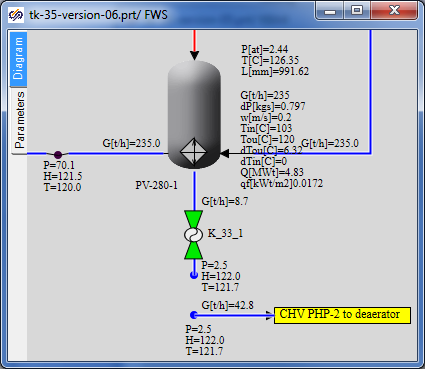


Figure 127. Connection of PHP-2 to Deaerator

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.

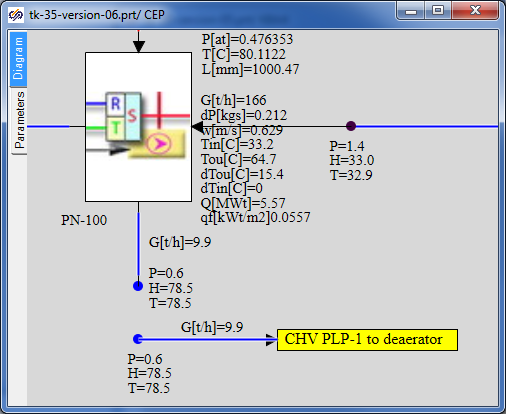


Figure 128. Connection of PLP-1 to Deaerator

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).

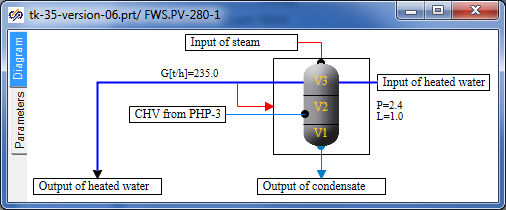


Figure 129. Addition of New Hole to PHP-2

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.

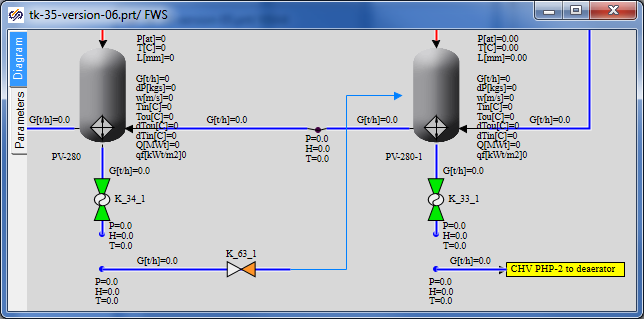


Figure 130. PHP-3 and PHP-2 Connection

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.

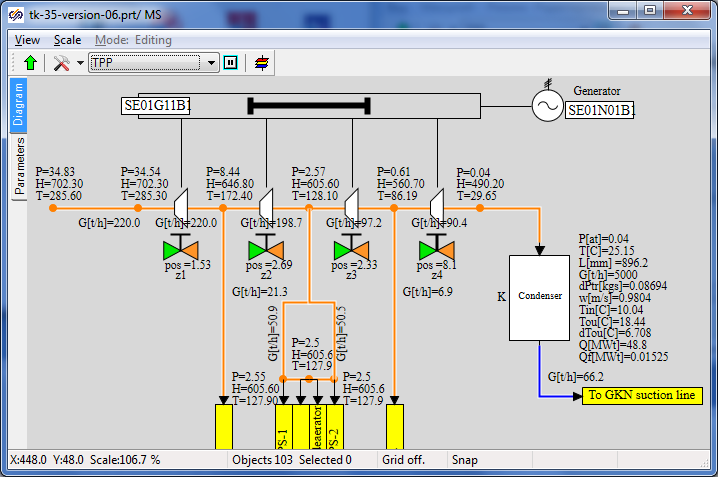


Figure 131. Stationary State after Connecting CHS to Deaerator

## 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”**.

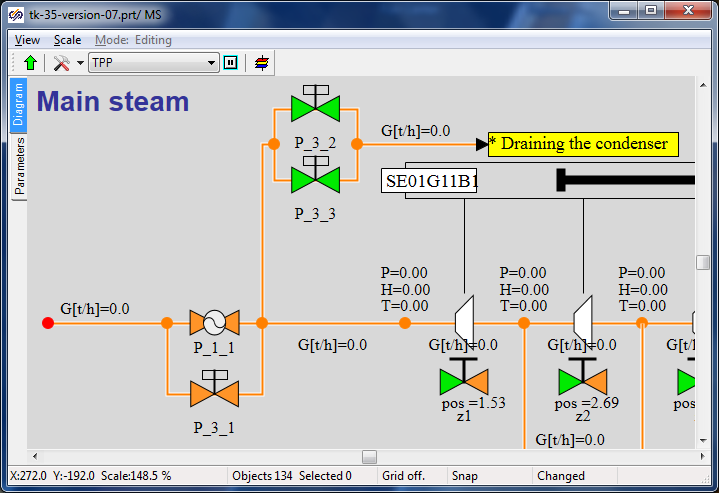


Figure 132. Draining to Condenser Model in Main Steam System

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

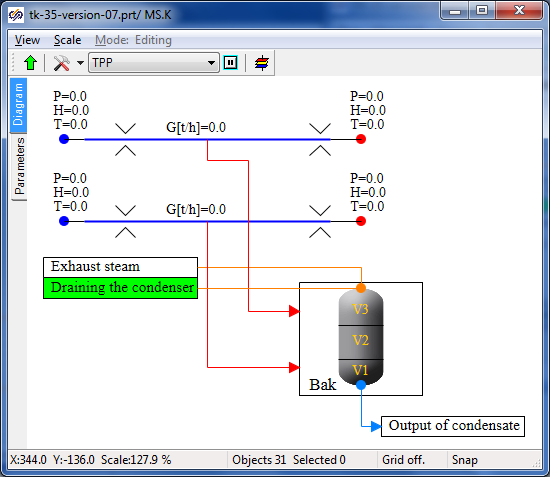


Figure 133. Connection to Condenser

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.

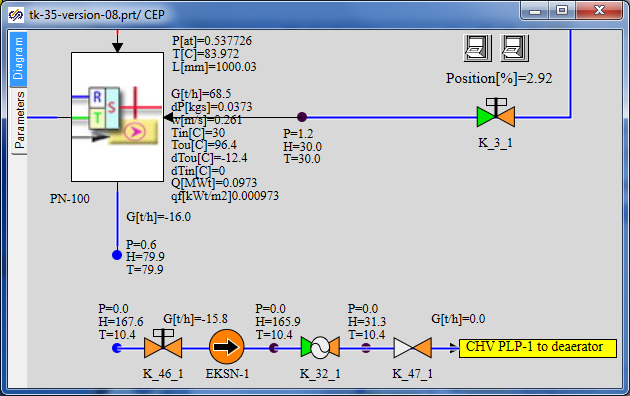


Figure 134. Structure of Introduction of pump EKNS-1 into PLP-1 model

### 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.

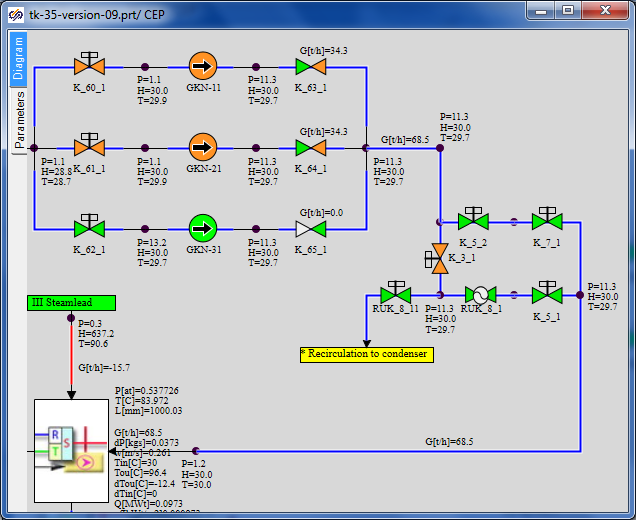


Figure 135. CEP Recirculation to Condenser

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)