# CAD/CAM System N-Ship+

## Version 5.0

# Module Part

# Creation of hull parts.

# Technological documentation

**NSHIP.00005.005-2025**

**User manual (nanoCAD)**

**Sheets 81**

**Saint Petersburg**

**2025**

ANNOTATION

The document is a reference manual for work with the module **Part** of the **N-Ship+** system in environment of graphical processor Platform nanoCAD. The module is aimed to form geometry of hull parts to prepare data for manufacturing and to issue technological documentation for parts

**N-Ship+** is informationally compatible with the systems **Ritm-Ship** (AutoCAD), **R-Ship+** (AutoCAD), **B-Ship+** (BricsCAD).

Document is designed for specialists who run **N-Ship+** system for the construction and technological preparation of the ship hull production and have practical experience of using nanoCAD or AutoCAD.

Contact data:

Mobile: **+7 921 7561226** (Nikolai Poleshchuk)

Email: **npol50@yandex.ru**

Web: **http://poleshchuk.spb.ru/cad/2016/nshipe.htm**

# CONTENTS

TERMS AND AGREEMENTS

This guide uses the following font agreements:

*Italic* – names of folders, files and extensions, additional text to graphical editor requests in commands;

**Bold** – names of modules and system components, menus, items, buttons and keys, commands in the dialog with graphical editor;

CAPITAL – names of layers, software commands and named objects.

For shortness everywhere in the document system N-Ship+ will be named N-Ship.

# 1 MODULE DESIGNATION

Module **Part** is targeted for forming graphical and textual data for sheet and profile parts and for technology of manufacturing. Module is functioning in interactive mode but some operations can be run in batch mode, e.g. generation of TNCs (technologic norming cards), or FPDs (forms of printed documents).

Module **Part** works with DB of project portion (it is called project_port, e.g. EN103_1). Before work user must run module **Bdata** and create database tables and structured project_port folder. First of all module **Bdata** must fill in parts table, called also *specification*, or *draw*. Parts table file is named *specp.dbf*.

Sheet and profile parts are described/generated by draws. Results for parts include DWG files with geometry and DB tables with textual data (area, work number of user, etc.).

Module has different language localizations. Language can be changed by menu item **BDATA > UI language**.

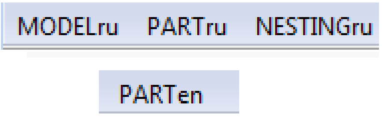
# 2 SCHEME OF USING MODULE

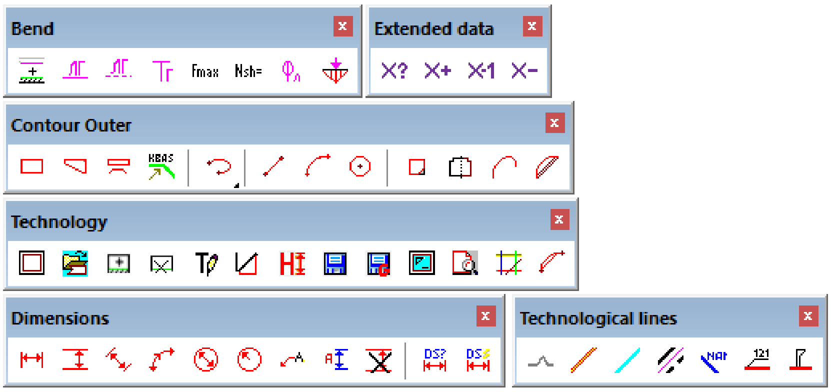
## Start of work with module Part

Before using module **Part** it is necessary to set current project_port, for example: EN103_1) with module **Bdata**.

Module has a drop-down menu, named **PART** with two symbols suffix denoting localization language: **PARTen** (English), **PARTru** (Russian), etc. (pic. 1). For shortness everywhere in the manual menu will be named without suffix (**PART**).

Module has toolbars too (pic. 2).

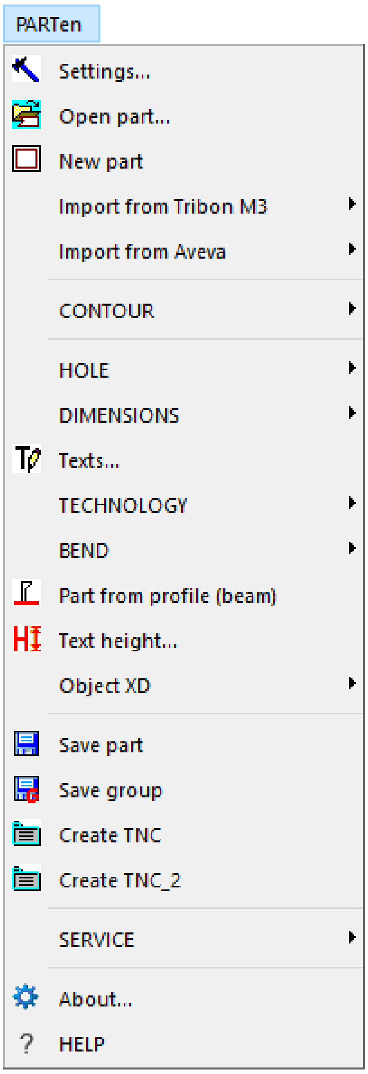



Pic. 1. N-Ship menu captions    Pic. 2. Toolbars of module **Part**

Drop-down menu PART

On pic. 3 there is a drop-down menu **PART** containing commands and submenus of module **Part**:




Pic. 3. Drop-down menu **PART**

**Note.** Commands of import from Tribon M3 and Aveva are described in the documents **Import of parts from system Tribon M3 (nanoCAD)** and **Import of parts from system Aveva (nanoCAD)**.

Command **Settings** calls dialog box **Set parts attributes** for setting features of part drawing decoration and some modes of module work.

Command **Open part** opens dialog **Set part** to open DWG geometry files for one part or several parts. Command **New part** creates new empty drawing, with necessary settings and layers for part.

Submenu **CONTOUR** includes commands for building outer contour line of the sheet part. For some kinds of parts there are automatic tools (rectangle, knee, etc.).

Submenu **HOLE** is designed for creation of inner and contour holes (both for cut and uncut holes).

Commands of submenu **DIMENSIONS** facilitate dimensioning part contour, placing objects on layer DIM.

Command **Texts** gives opportunity to create manufacturing texts in the area of part.

Submenu **TECHNOLOGY** is intended for technological objects like lines, allowances, chamfers and bevels as well as texts. Submenu contains nested submenus **Lines**, **Allowance**, **Chamfers and bevels**.

Commands of submenu **TECHNOLOGY**>**Lines** allow to draw technological lines inside part contour (marking, girder attachments etc.). Commands of submenu **TECHNOLOGY**> **Allowance** are intended to create allowance to part outer contour and to inner holes, and can remove allowance if necessary. Commands of submenu **TECHNOLOGY> Chamfers and bevels** are designed to create technological texts for chamfers and bevels in a special form with important extended data.

Submenu **Object XD** contains commands for extended data operations (reading, adding, etc.).

Commands of submenu **BEND** allow drawing lines and texts for bending, as well for bending templates.

Command **Part from profile (beam)** accumulates operations for profile part creation (including developed sketches for bended parts) and a command for drawing profile transverse section.

Command **Text height** calls dialog box for changing heights of texts inside part area.

Command **Save part** finalizes calculation of part parameters and stores them in DB, saving geometry DWG file in the folder *Dwg* of current project_port. Command **Save group** allows to run controlling calculations for a group of parts and save their parameters in DB.

Commands **Create TNC** and **Create TNC_2** open window for selection of part positions for which forms of part document (FPD, or TNC) must be generated.

Submenu **SERVICE** has service commands (check contour, etc.). Item **About** calls window with info data about module version and developers names. Command **HELP** displays help window for module **Part**.

# 3 PART FILE STRUCTURE

Result of building part geometry is a DWG file with contour lines, text and dimension entities. Some objects can have extended data (xdata). All the information will be later used by module **Nesting** and allows to place sheet parts inside nesting maps (plates) and to generate CNC programs for part manufacturing (cutting, marking, etc.).

## Part layers

DWG file for part is structured by layer names. Layer names are fixed:

KBAS, layer for base outer contour of the part (without contour holes and allowances);

KNOTCH, layer for contour holes, or knotches;

KHOLE, layer for inner holes;

KHOLEN, layer for uncut holes;

PRIPSB, layer for elements of allowance designations;

KONTUR, layer for the source contour part that is excluded – for example, deleted allowance edge (marked with green color);

MARK, layer for part label and orientation texts;

FASKA, layer for chamfer elements;

LASKA, layer for bevel elements;

SVERL, layer for drilling elements;

TIPDET, layer for bending elements;

SKR, layer for elements of rounding free edges;

STAMP, layer for stamp operation elements (part or holes);

SLED, layer for girder attachment lines;

KALL, layer for assembled outer contour, with inserted contour holes and allowances (final outer contour object is automatically created while saving part geometry);

KHALL, layer for assembled inner contour, with allowances (final inner contour object is automatically created while saving part);

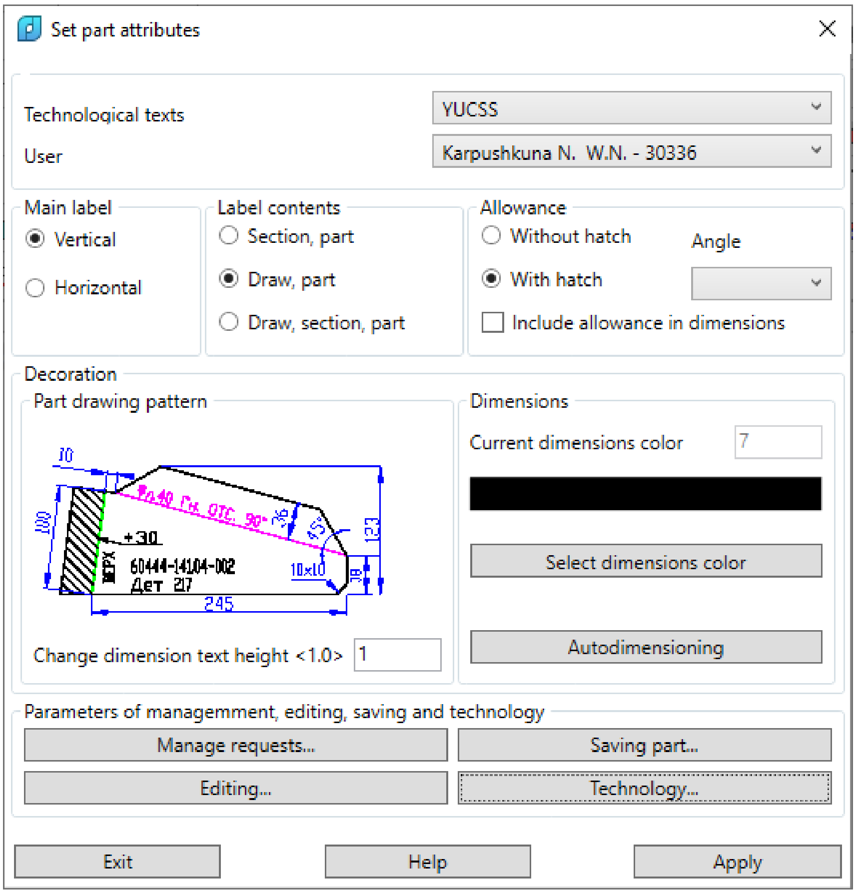
RAZM, layer for marking lines to be transferred to nesting map and including into CNC, or for girder lines on parts manufactured from panel materials.

Layer KBAS is a main one. It must contain base outer contour entity of 2D-POLYLINE. Moreover it must be closed, unique and be situated on zero elevation in WCS.

**Note.** For creating part from the zero state (from the very beginning) it is strongly recommended to apply command **New part** (see pic. 3). It creates all the part layers and it **prevents situation** when program stops because **all the drawings were closed** (it is critical for LISP functions).

# 4 SETTING PART ATTRIBUTES

On pic. 4 there is a dialog box **Set part attributes** called with the menu item **PART > Settings**.




Pic. 4. Dialog box **Set part attributes**

### Main settings

In the line **Technological texts** there is a folder name (usually connected with shipyard) for saving technological attributes. In the drop-down list **User** reflects name of the active user from the users list created earlier in project_port DB users table.

Down there are three areas: **Main label**, **Label contents**, **Allowance**.

Labelling part can be done in one line (**Horizontal**) or in two lines (**Vertical**). Label contents can include corresponding visible attributes, defined by radio column **Label contents**.

After applying allowance to part contour segment it can be hatched or not, that is defined by radio buttons in the area **Allowance**. Setting checkbox **Include allowance in dimensions** ensures creating part dimensions with allowance included. Drop-down list **Angle** defines hatch angle for allowance zone (default is 45 degrees).

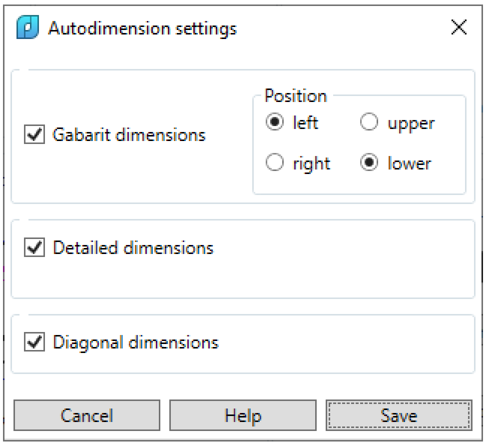
Area **Part drawing pattern** shows slide with part sketch display using current label type.

In the area **Dimensions** button **Select dimensions color** provides setting new color for dimension elements (number from 1 to 255). Button **Autodimensioning** changes settings in a separate dialog for dimensions created in an automatic dimensioning command.

Area **Parameters of management, editing, saving and technology** contains four buttons for calling dialog boxes for some types of settings.

### Autodimensioning

Button **Autodimensioning** opens dialog box **Autodimension settings** (pic. 5).



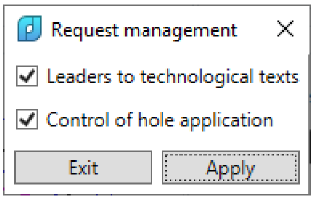

Pic. 5. Dialog **Autodimension settings**

Checkboxes **Gabarit dimensions**, **Detailed dimensions**, **Diagonal dimensions** define building correspondent groups of dimensions for part contour from layer KBAS in automatic mode. For gabarit dimensions there is also an area **Position** with radio buttons to control dimension placement.

Settings are saved with the button **Save** and then are used by the command **PART > DIMENSIONS > Autodim**.

### Requests management

Button **Manage requests** opens dialog box shown on pic. 6.




Pic. 6. Window **Request management**

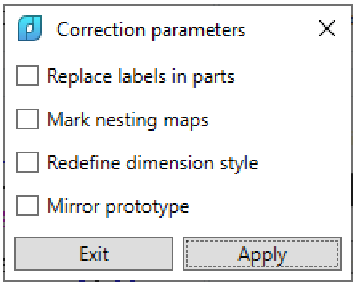
There are two checkboxes in the window. Setting checkbox **Leaders to technological texts** provides drawing of leader line to texts like chamfer, bevel, allowance, etc.

Checkbox **Control of hole application** allows to stop process of saving part, for visual control of contour holes attachment.

Button **Apply** is intended for saving settings of this window. Button **Exit** closes the window without saving.

### Editing

Button **Editing** of settings dialog (see pic. 4) opens window **Correction parameters** (pic. 7).




Pic. 7. Window **Correction parameters**

If checkbox **Replace labels in parts** is set then during group resaving labels will be changed to new form, from current settings.

If checkbox **Mark nesting maps** is set then when part is resaved then it gets sign about necessary correction of nesting maps with this part.

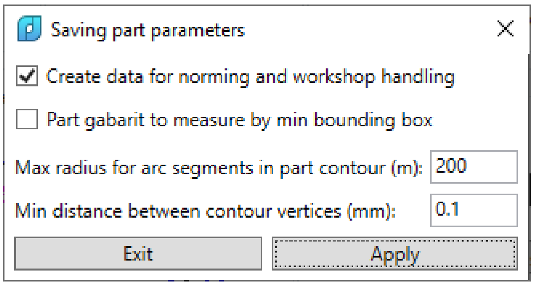
If checkbox **Redefine dimension style** is set then while opening part geometry file its dimension style is replaced.

If set checkbox **Mirror prototype** then if new part will use an old one as prototype then prototype geometry will be automatically mirrored.

Button **Apply** saves these settings. Button **Exit** closes window without saving settings.

### Saving, data for norming

Button **Saving part** of settings dialog (see pic. 4) calls child window **Saving part parameters** (pic. 8).




Pic 8. Window **Saving part parameters**

Checkbox **Create data for norming and workshop handling** allows automatically append entities with additional norming information (edge length with chamfer, etc.). Checkbox can be disabled depending on purchasing system conditions.

Activating checkbox **Part gabarit to measure by min bounding box** requires taking gabarits using bounding box around part outer contour.

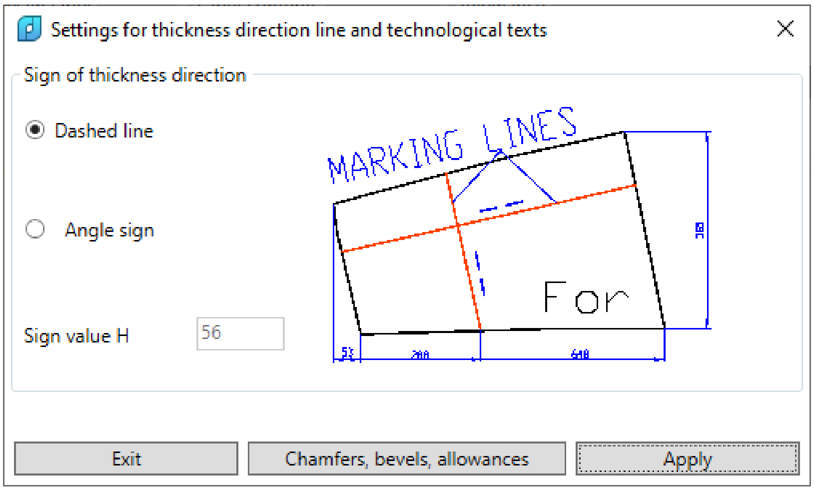
In the field **Max radius for arc segments in part contour (m)** there is radius value that is not recommended for changing.

Similar recommendation can be extended to the field **Min distance between contour vertices (mm)**.

Button **Apply** serves for saving these settings.

### Technology settings

Button **Technology** of settings window (see pic. 4) calls dialog box **Settings for technological texts and thickness direction line** (pic. 9).

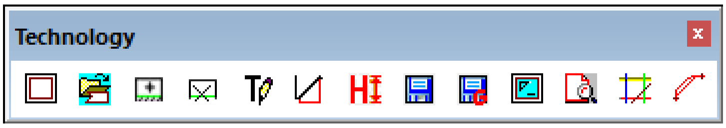



Pic. 9. Window **Settings for technological texts and thickness direction line**

Activating radio button **Dashed line** in the area **Sign of thickness direction** defines thickness direction sign as two dashes. Radio button **Angle sign** provides insertion of thickness sign into the line, in the form of two lines angle, with the height value, entered in the field **Sign value H**. Button **Chamfers, bevels, allowances** calls special window of settings for multitexts for chamfers, bevels and allowances. Work with it is discussed hereinafter.

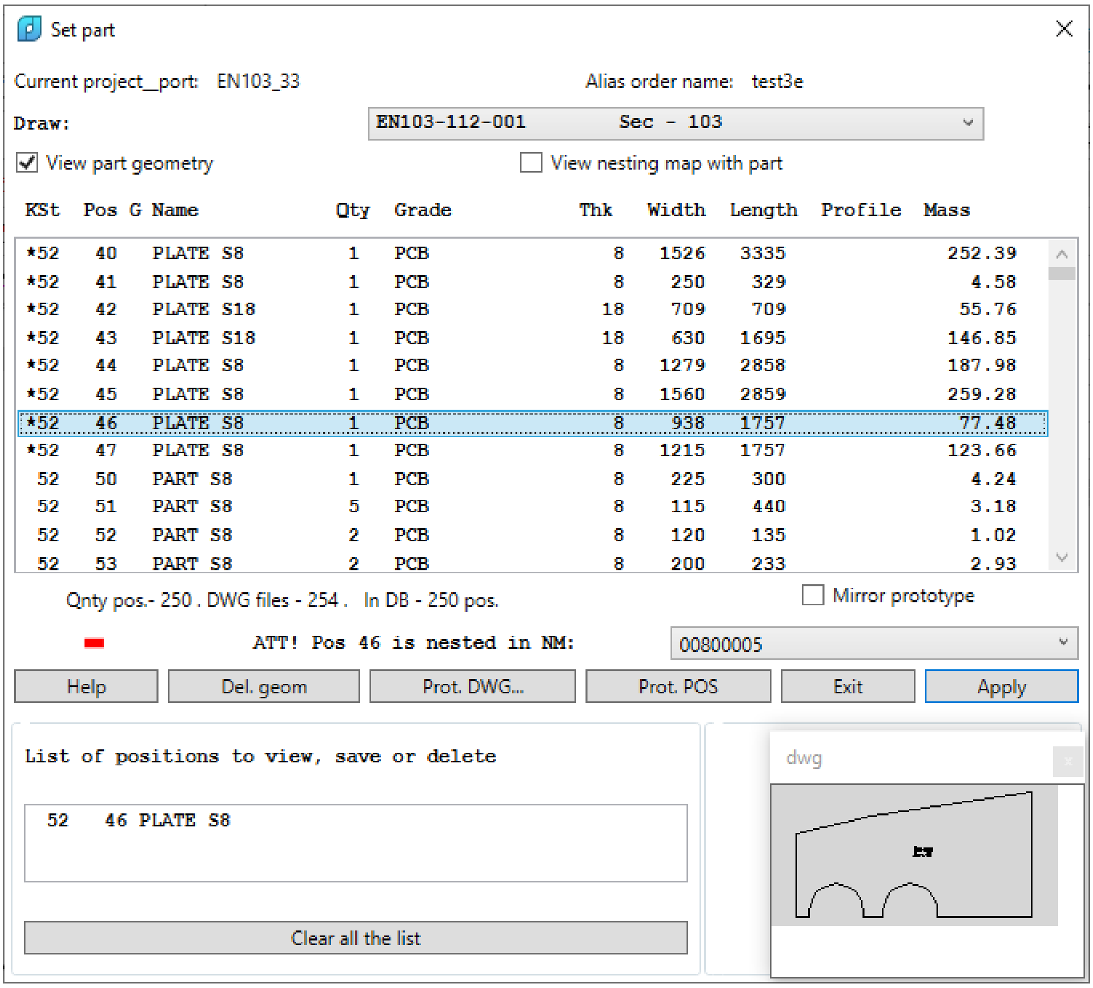
## 5 OPENING PART DRAWING

To open on screen the geometry file of earlier saved part (for editing, viewing, resaving, etc.) there is a menu item **PART > Open part** or button  of the **Technology** toolbar (pic. 10).




Pic. 10. Toolbar **Technology**

Command **Open part** calls dialog box **Set part** shown on pic. 11.




Pic. 11. Window **Set part**

It enables selecting specification (part positions list) position and loading on screen part sketch drawing. It is also used for creating a new part using an old part as prototype. Upper line of the dialog box shows current project, order.

Dialog displays current project_port and order designation. Drop-down list **Draw** contains list of order draws (specifications), for selection of other draw. After draw selection parts table in the central zone of the window refreshes.

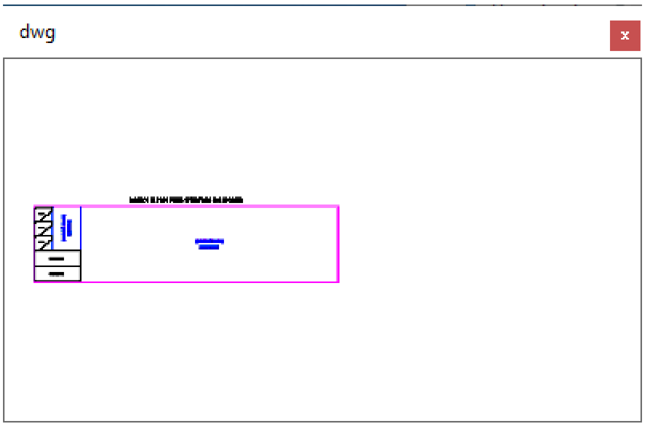
In the positions list parts having DWG geometry files (in the *Dwg* folder of the current project_port) are marked with asterisk (*). After click on part line its name is copied to the lower **List of positions to view, save or delete**. And at the same time in the lower right corner there appears raster part image (if the part has geometry and box **View part geometry** is checked).

On clicking button **Apply** window closes and procedure of loading DWG file of selected position starts. If lower list includes more than one line then every part is shown in its own document window.

If button **Del.geom** is pressed then DWG file of marked part is being deleted (needs confirmation).

Buttons **Prot. DWG** and **Prot. POS** are created for calling prototype DWG file to become geometry file of selected position. Prototype can be defined by selecting DWG file or by other part position number. If checkbox **Mirror prototype** in the dialog **Correction parameters** (see pic. 7) is set, then prototype geometry is being mirrored (it is useful for creating parts for different sides).

If checkbox **View nesting map** with part is set then to the right of the windows there appears a window displaying DWG file of the map in which the part was nested (pic. 12). List of all the maps is filled in the list after the text **ATT! Pos <No> is nested in NM:**.



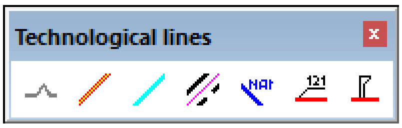

Pic. 12. Window with nesting map including required part

Clicking button **Clean all the list** clears the lower list (selection).

Button **Exit** closes window with no actions.

## 6 CREATION OF PROFILE PART

Operations with profile parts is done with complex command **PART > Part from profile (beam)** or with button  of toolbar **Technological lines** (pic. 13).




Pic. 13. Toolbar **Technological lines**

Sequence of operations

Command starts with these options in the command line:

*Mode [Section/Line/Curve/E-sketch/Parameters/eXit] <eXit>*.

User must select continuation mode with an option (the capital letter should be used while keyboard input):

- **Section**, drawing transverse section of profile;

- **Line**, building straight profile part contour, with picking two points for attachment line in model (in WCS);

- **Curve**, building curved profile part contour, with picking attachment line (2D-POLYLINE in WCS) and two points;

- **E-sketch**, building in new drawing 2D sketch of developed profile part for the prevoius contour of curved/straight part in model (options **C** and **L**);

- **Parameters**, output of profile part parameters saved in contour extended data after options **L** and **C**;

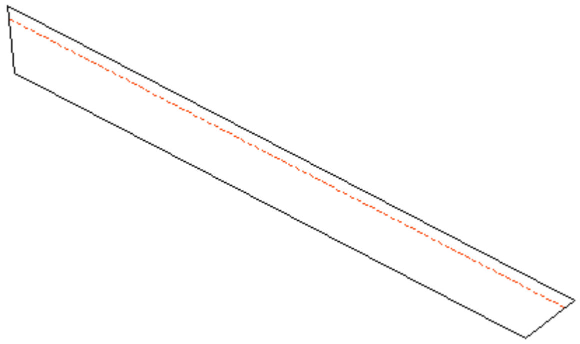
- **eXit**, stopping command with no actions.

Options **L** and **C** possess an opportunity to go to sketch creation immediately after building contour in model. Sketch is drawn in a new drawing. As most of profile parts are narrow and long then during sketch creation there is offered an option to scale sketch for more convenient work in the workshop.

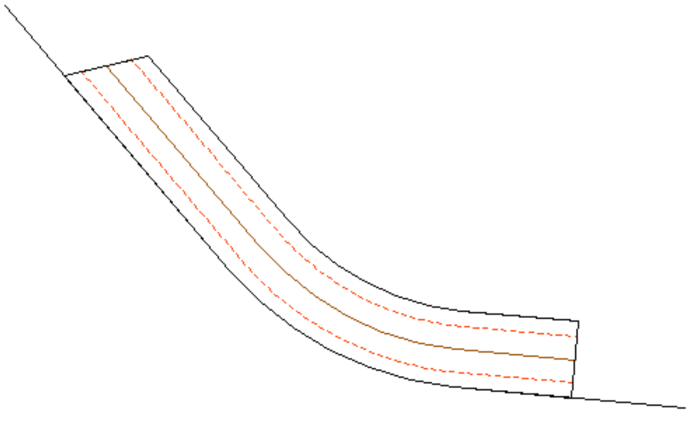
After drawing profile part sketch it is necessary to append other decoration elements (cuts, holes, allowances, chamfer texts, etc.) with items of menu **PART** and to save sketch with menu item **Save part**. DWG file with 2D sketch geometry is stored in the subfolder *Dwg* of current order and textual attributes are saved in DB table *specp.dbf*.

Sample results of part calculation

On pic. 14 and 15 there are sample results of building contours of straight and curved profile parts in model WCS.

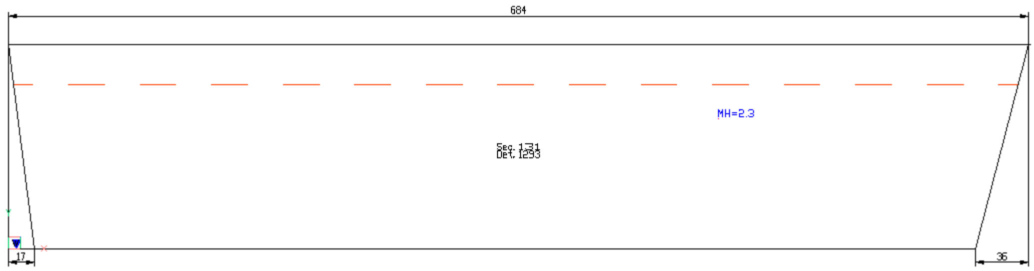


Pic. 14. Contour of straight profile part in model



Pic. 15. Contour of curved profile part in model

On pic. 16 there is a sample result of drawing 2D sketch of profile part. If part in model has a curvilinear attachment line then development is calculated (length is taken from neutral line, or neutral layer).




Pic. 16. Sample result of 2D sketch

Draw section

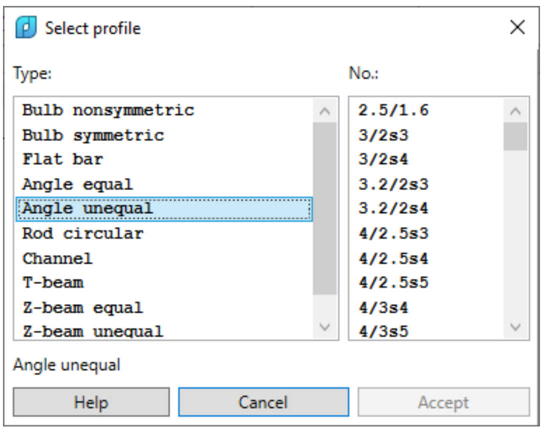
Operation of creating section is an auxiliary one and is executed with option **Section** of command **Part from profile (beam)**. Next request:

*Current profile parameters: type=72, No=SH40*

*Select different profile? [Y/N] <N>:*

Program remembers number of the last profile (in example SH40 for type 72, i.e. channel). Profile type have thr following numbers: 30 – bulb nonsymmetric, 31 – bulb symmetric, 40 and 85 – rod circular, 50 – T-beam, 60 – double T, 61 – Z-beam symmetric, 62 – Z-beam nonsymmetric, 70 – angle equal, 71 – angle unequal, 72 – channel, 91 – flat bar. Reply **N** means work with the previous profile.

To change profile it is necessary to select **Y**. Dialog box **Select profile** opens (pic. 17).




Pic. 17. Dialog box **Select profile**

In the list **Type** it is necessary to select profile type, and in the list **No.** to select number. After selection of number click button **Accept**. Window will close and request will be output:

*File np_prof01.ini saved.*

*Current profile parameters: type=30, No=9*

*Mounting parameters: flange orientation = right, web thickness = right.*

*Do you wish to change? [Y/N] <N>:*

Section will be drawn with using web direction and flange orientation relative insertion point and insertion axis. If suggested data do not fit user then enter **Y**. Next:

*Flange orientation [0-right/1-left] <0>:*

Select required value (0 or 1). Next:

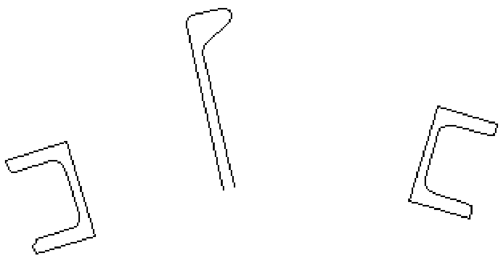
*Web thickness direction [0-right/1-left/2-symmetric] <0>:*

Here are allowed three variants of web direction relative insertion axis. Final requests:

*Insertion point <exit>:*

*Angle (point) of web inclination <exit>:*

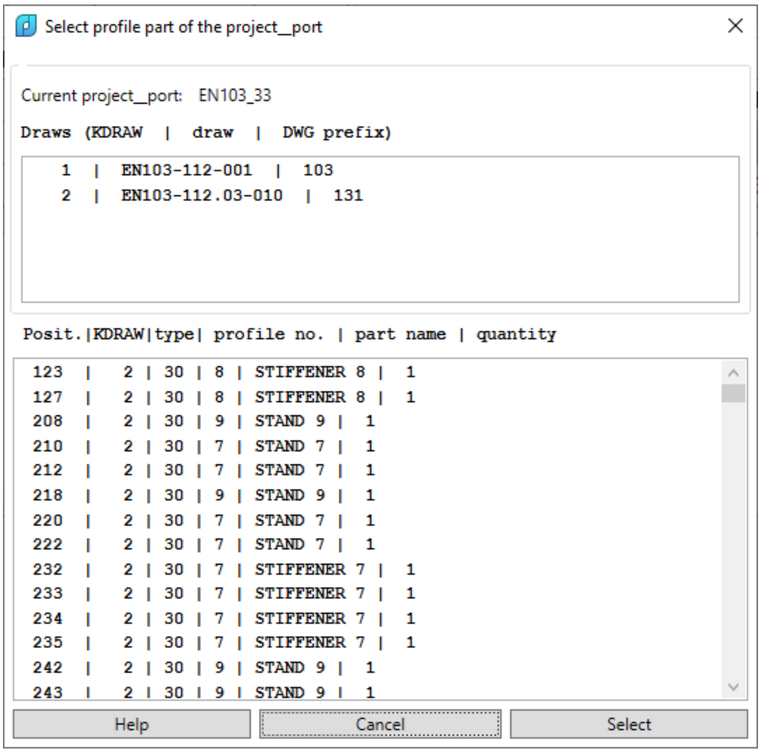
It is necessary to pick two points (for insertion and web axis direction). Sample results are shown on pic. 18.



Pic. 18. Profile sections

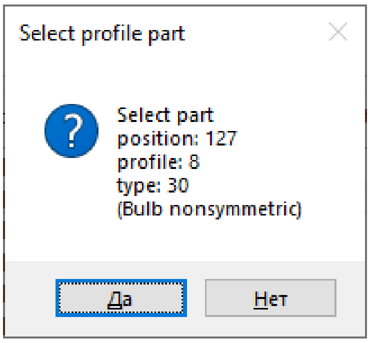
Straight profile part

Option **Line** of command **Part from profile (beam)** corresponds to calculation mode for straight profile part in model. After selection of **L** dialog box **Select profile part of the order** (pic. 19) opens.




Рic. 19. Окно **Select profile part of the project_port**

In the bottom side of the window there is list of profile parts (only) for the current project_port. Each line displays position number, KDRAW of draw, profile type, profile number, part name and quantity of parts for this position. Select position that is needed for future work. After pressing button **Select** there a request is generated, it requires selection confirmation (pic. 20).




Pic. 20. Request **Select profile part**

If click **No** (Нет), then user returns to the dialog box **Select profile part of the project_port** for reselection. To confirm selection click **Yes** (Да). Next:

*Creating straight profile part by two points...*

*File np_prof01.ini saved.*

*Working in model, in world coordinate system (WCS).*

*Building straight line of profile part.*

*Z value of the first point will fix objects elevation in WCS.*

*Point of butt 1:*

Pick the point of first butt for the future line segment to be used as part attachment line. Building is done in WCS. It is the point on butt 1. If the point 1 is given with the nonzero value of Z coordinate then default building will be done with this nonzero elevation. Next request:

*Set elevation value Z=0.0*

*Do you wish to change elevation? [Y/N] <N>:*

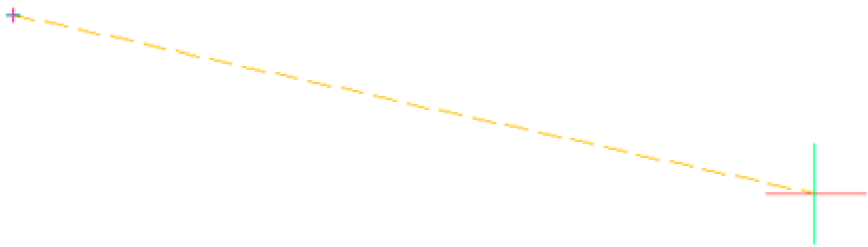
At this moment user is still able to change elevation (Z), taken from the point 1. To change press **Y**. Request in this case:

*New elevation for Z <0>:*

Any real number (positive or negative) can be entered as a reply. Next:

*Point of butt 2:*

Screen will show rubber band, facilitating point input (pic. 21).



Pic. 21. Input of first point for butt 2

After second point input the part attachment line will be fixed. Going to butt inclinations.

*Defining butt lines.*

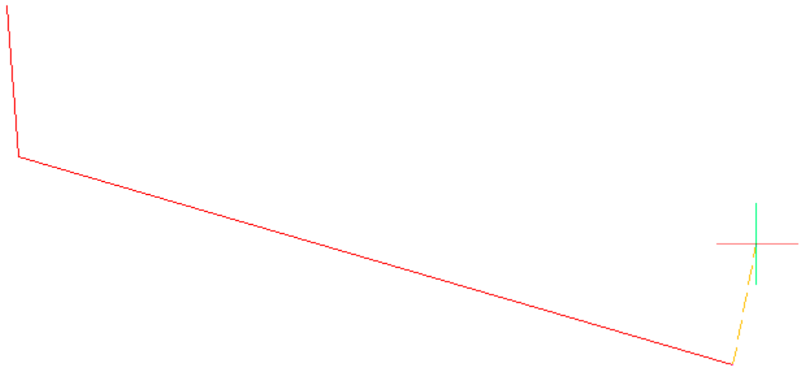
*Second point of butt 1:*

Pick the point that will fix inclination of butt 1 boundary line (only angle will be taken from this line). If butt must be normal then it can be amended later, now pick something close to desired result.

After the first butt there is a question about butt 2:

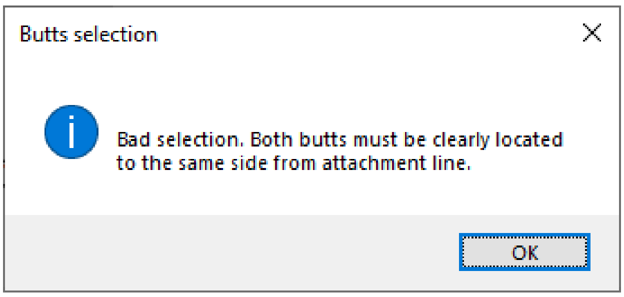
*Second point of butt 2:*

On pic. 22 there is a moment of entering second points shown.



Pic. 22. Input of second point for butt 2

Input of second point for butt 2 is additionally verified for proper location: points of butts 1 and 2 must be placed to the same side from attachment line. If error is found then message is displayed (pic. 23).




Pic. 23. Error in input of second point for butt 2

User should repeat command with option **Line**.

If input is correct then there will be additional questions enabling to set exact normal direction of butts:

*Create butt 1 exactly by normal? [Y/N] <N>:*

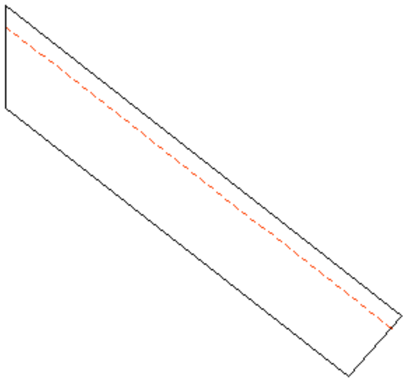
*Create butt 2 exactly by normal? [Y/N] <N>:*

If **Y** replied then butt direction(s) will be modified up to normal.

Next request is connected with display mode for flange lines: with visible or hidden linetype. Nonsymmetric bulb and unequal angle have only one flange line (upper), channel has two lines (upper and lower). For other profile types there is no similar request, only visible linetype is applied.

*Flange linetype in model [F (continuous)/B (dashed)] <F>:*

Reply **F** refers to visible line (type Continuous), **B** refers to hidden line (type DASHED1). Linetype is applied to the current model view (sample result is shown on pic. 24).



Pic. 24. Part view in model

In the extended data of outer contour there are saved parameters of part and profile type.

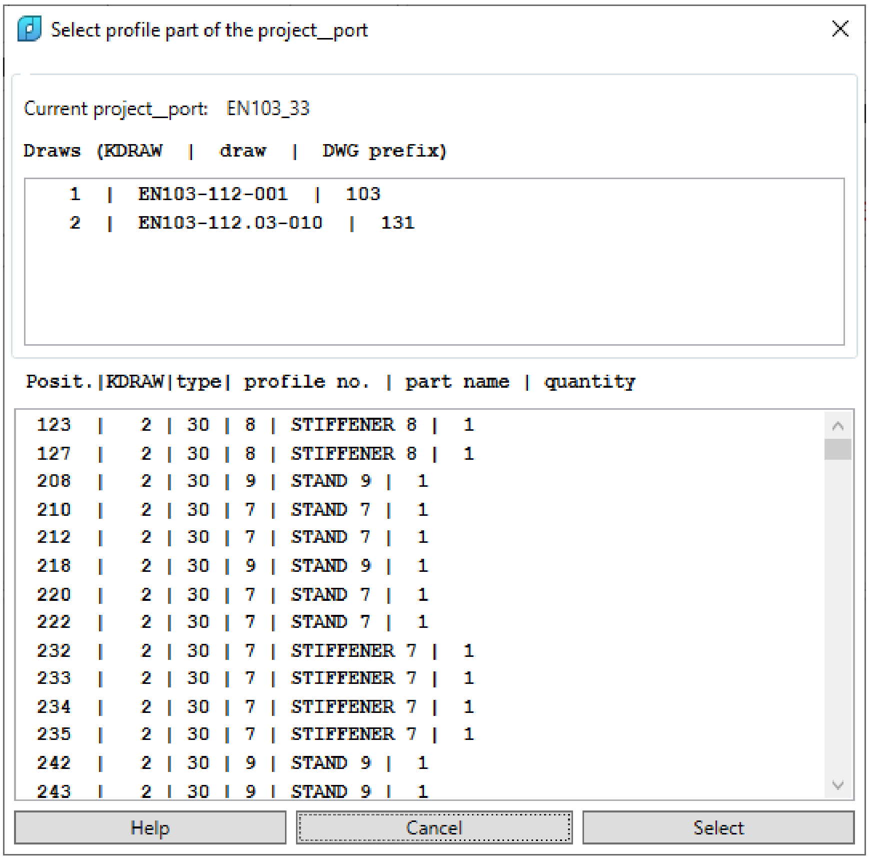
Curved profile part

Option **Curve** of command **Part from profile (beam)** corresponds to calculation mode for non-straight profile part in model. First request:

*Creating profile part in model by curved attachment line...*

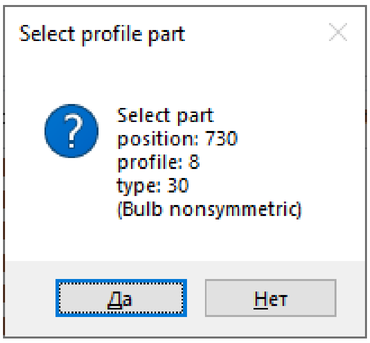
*Select open attachment line built in WCS (2D polyline):*

User must select open 2D polyline (LWPOLYLINE or 2D-POLYLINE), segment of which will be taken as attachment line of profile part. After selection dialog box **Select profile part of the order** (pic. 25) opens.




Pic. 25. Dialog **Select profile part of the order**

This window was earlier used in the procedure of building straight profile parts (see pic. 19). In the bottom side of the window there is a list of profile parts included in parts specification for current order. For each part there are position number, part draw KDRAW, profile type, profile number (name), part name and quantity of parts for this position. Select position (part), with which work will be run. After pressing button **Select** there is a request for selection confirmation (pic. 26).




Pic. 26. Window **Select profile part**

To confirm click **Yes** (Да). Next:

*File np_prof01.ini saved.*

*Creating curved part in WCS, model.*

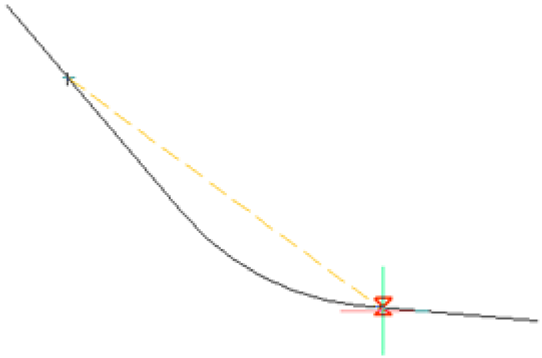
*Point of butt 1, on attachment line:*

*Set elevation value Z=0.0*

User must pick the first point on selected 2D polyline, it will become the first point of part attachment line. For convenience object snap Near is set on. Next request:

*Point o butt 2, on attachment line:*

On screen there is a rubber band, facilitating point input (pic. 27).



Pic. 27. Input of first point for butt 2, on attachment line

After input of point fixing second end of the attachment line, program goes to defining butt inclinations.

*Defining butts.*

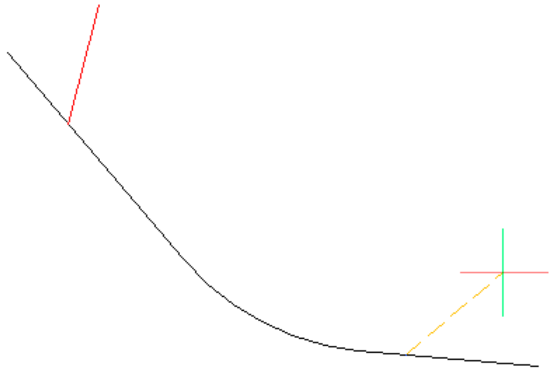
*Second point of butt 1:*

As a reply it is necessary to pick point defining inclination angle of butt 1 (only angle, because real part boundary will be calculated by profile height). If butt must ne normal to attachment line then user must pick as close as possible and it can be amended to true normal later.

After butt 1 then there is a request concerning butt 2:

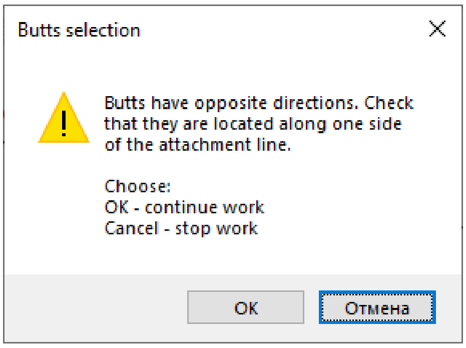
*Second point of butt 2:*

Pic. 28 reflects moment of second points input.



Pic. 28. Input of second point for butt 2

Input of second point for butt 2 is additionally verified for good location: points of butts 1 and 2 must be placed to the same side from attachment line. If error is found then message is displayed (pic. 29).




Pic. 29. Warning for suspicious input of second point for butt 2

If input is correct (**OK**) then there will be additional questions enabling to set exact normal direction of butts:

*Create butt 1 exactly by normal? [Y/N] <N>:*

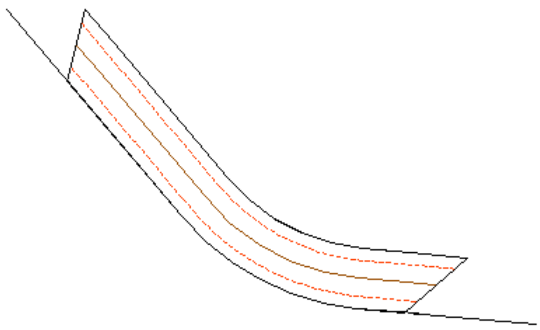
*Create butt 2 exactly by normal? [Y/N] <N>:*

If **Y** replied then butt direction(s) will be modified up to normal.

Next request is connected with display mode for flange lines: with visible or hidden linetype. Nonsymmetric bulb and unequal angle have only one flange line (upper), channel has two lines (upper and lower). For other profile types there is no request concerning flange line type.

*Flange linetype in model [F (continuous)/B (dashed)] <F>:*

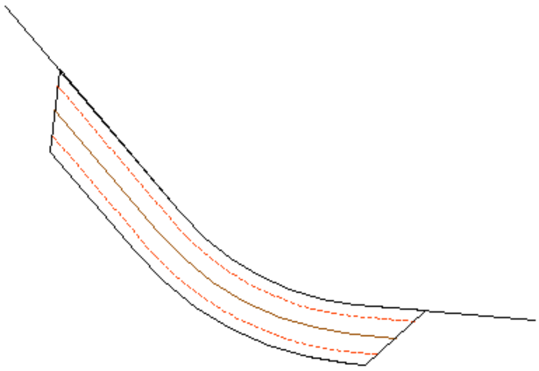
Reply **F** refers to visible line (type Continuous), **B** refers to hidden line (type DASHED1). Linetype is applied to the current model view (sample result is shown on pic. 30, when part is located to the left (upper) side of polyline direction).



Pic. 30. Curved part view in model (left)

With brown color there is shown a line of part neutral layer.

On pic. 31 there is a sample case when part is located to the right side of attachment line direction.



Pic. 31. Curved part view in model (right)

In the extended data of outer contour polyline there are saved parameters of part and profile type. On finish of model building there is a suggestion to automatic transfer to 2D sketch creation in a new drawing:

*Create horizontal 2D sketch, with possible vertical scaling? [Y/N] <Y>:*

If **Y** then there is automatically called the program for drawing sketch of developed curved part. Process of drawing sketch is discussed hereinafter.

### Two dimensional sketch of profile part

Option **E-sketch** of command **Part from profile (beam)** corresponds to mode for calculation of profile part sketch using earlier built contour in model. The same mode starts when user gives affirmative reply to the question after building part contour in model (both for straight and curved parts).

Building sketch is executed in a new DWG drawing. If sketch mode was run independently from building part contour in model, then there is a request:

*Creating sketch of the developed profile part using model data...*

*Select closed polyline of the profile part contour:*

It is necessary to select in model closed polyline for outer contour of profile part, in WCS.

Next there are displayed current data values:

*Creating horizontal 2D profile part sketch in a new drawing.*

*Region="3"*

*Block="3"*

*Section="131"*

*KDRAW="2"*

*Draw="EN103-112.03-010"*

*Position="243"*

*PartName="STAND 9"*

*Material="A40S"*

*Thickness="5.5"*

*Prefix="131"*

*MatCode="00309453128"*

*Route="57"*

These data will be used for saving part in DB. Next there is an information about proportion between length and height for the part contour in model, for example:

*The proportion length/height for the part = 6.3.*

If proportion has value greater than 4 then part is counted as long and narrow and its usual image with scale 1:1 becomes bad to read. In this case program offers artificially increasing of height in sketch geometry:

*This is a long part.*

*Enter new proportion length/height for sketch or [N use old] <5.0>:*

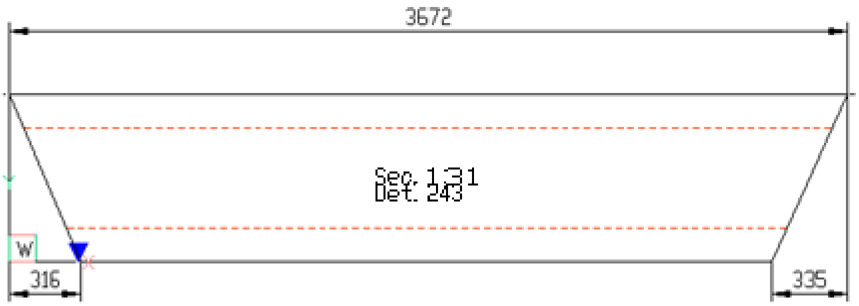
Here are options:

- preserve true proportions (option **N**);

- accept default proportion 5:1 (press Enter key);

- enter your own proportion (positive number).

The most usable proportion is default 5:1, it is activated with pressing Enter or mouse right-click. After selection of new value for length/height in sketch, there is drawn and being dimensioned part sketch in a new DWG file (pic. 32).




Pic. 32. Sketch for profile part

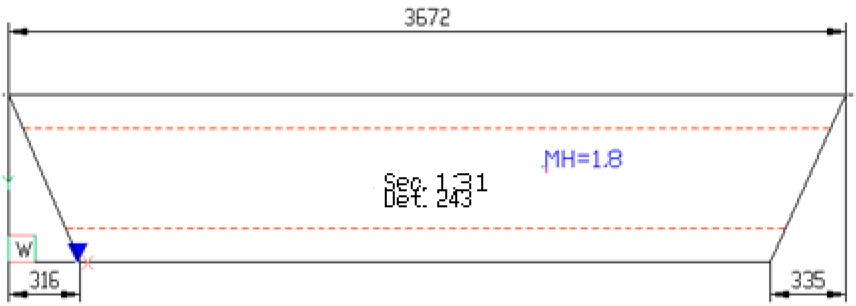
Automatically there is inserted in the center block of label mark, in the form set by current part settings (see par. 4). Label language is taken from the current interface localization setting. Linetype is the same as in model part view (see pic. 30). Near the butt that was first in part definition, there is a blue triangle sign. Next:

*Sketch scale for height MH=1.836.*

After entering new proportion between length and height (before it was equal to 1), therefore program offers to insert explanatory text:

*Insertion point for text "MH=1.8" <cancel>:*

The text informs whether the part sketch uses true or conditional dimensions. If press Enter then text will not be inserted (cancel). If pick a point then sketch will be appended with blue additional text (pic. 33).

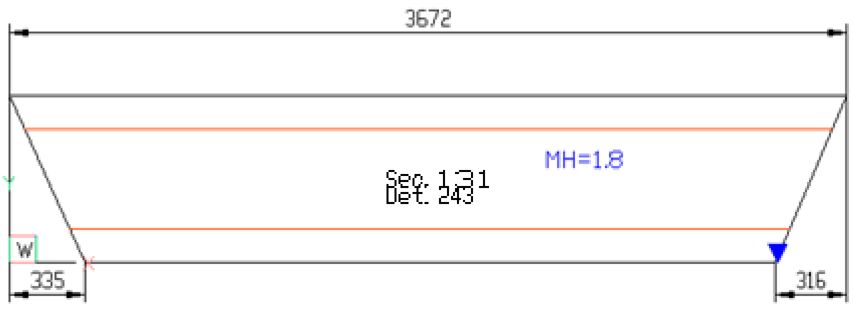



Pic. 33. Profile part sketch with new scale by height

Next request:

*Mirror sketch relative to vertical axis? [Y/N] <N>:*

Reply **Y** means that sketch must be mirrored relative to vertical axis. With this the first butt will go to right side and flange line can change its linetype in comparison with the source part view (pic. 34).




Pic. 34. Sketch after mirroring

Created part sketch must be saved into DB with command **PART > Save part**. Otherwise user may go adding other elements (scuppers, bending templates, etc.).

Output of part parameters saved in xdata

While building profile part outer contour there are formed extended data (XD) with parameters. These parameters include:

- order name (in the form project_portion);
- KDRAW of the draw (specification), including the part;
- position number;
- code of material type (profile type);
- profile number;
- profile height, mm;
- center of gravity ordinate for profile transverse section (ordinate is used in calculation of neutral layer for development);
- full part length (with slopes);
- part length without slopes;
- first slope, mm (by the first butt);

- second slope, mm (by the second butt);

- linetype to draw flange in model;

- part orientation code in model relative to attachment line (left or right along line direction);

- token of straight or curved part.

If in command **Part from profile (beam)** after the following request

*Mode [Section/Line/Curve/E-sketch/Parameters/eXit] <eXit>*

user selects option **Parameters** and picks contour in model, then there will be output values of the main eight parameters:

*Profile type: 30 (Bulb nonsymmetric)*

*Profile no.: 9*

*Profile height: 90.0*

*Y for profile section center of gravity: 56.5*

*Order: EN103_1*

*KDRAW of the draw: 2*

*Part position no.: 243*

*Length of part development: 386.9*

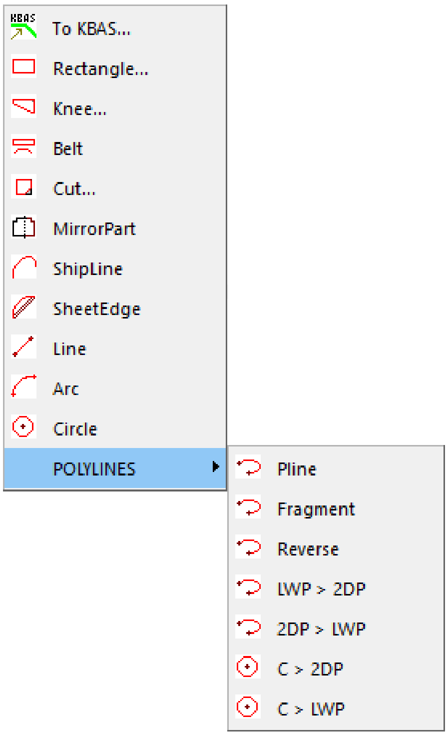
In the full scope part extended data can be got with menu item **PART > Object XD > Show XD**. For example:

*((-1 . #<Entity name: 000002464EF773B0>) (0 . "POLYLINE") (5 . "4E7") (330 . #< Entity name: 000002464EF87230>) (100 . "AcDbEntity") (67 . 0) (410 . "Model") (8 . "KBAS") (62 . 7) (48 . 100.0) (100 . "AcDb2dPolyline") (66 . 1) (10 0.0 0.0 0.0) (70 . 1) (40 . 0.0) (41 . 0.0) (71 . 0) (72 . 0) (73 . 0) (74 . 0) (210 0.0 0.0 1.0) (75 . 0) (-3 ("XDPROFILE" (1000 . "***PROJECT_PORT=BBBBB_2 KDRAW=1 POZ=427 KVM=30 PRF=9 H=90.0 YC=56.5 D=2387.6 D0=2387.6 D1=-32.2 D2=-102.5 LT=0 ZD=1 UB=0****"))))*

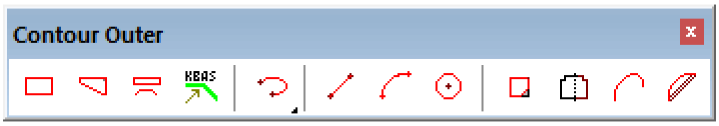
## 7 CREATION OF SHEET PARTS

### Outer part contour

Forming outer contour for the part, included into specification table, but still having no geometry is done with submenu **PART > CONTOUR**, shown on pic. 35 and toolbar **Contour Outer** (pic. 36).



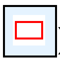

Pic. 35. Submenu **CONTOUR**

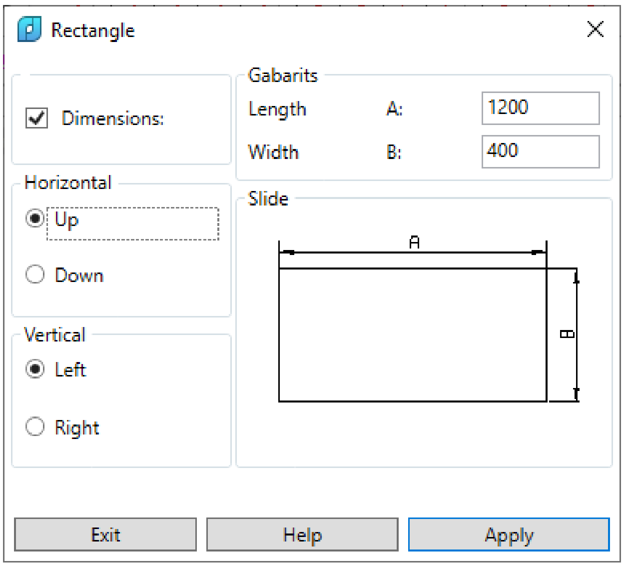



Pic. 36. Toolbar **Contour Outer**

Outer contour must be a closed object of type 2D-POLYLINE and must be located on layer KBAS.

Rectangle

Command **Rectangle** () is intended for building parts having rectangular form. Command opens dialog box shown on pic. 37.




Pic. 37. Window for rectangular part parameters

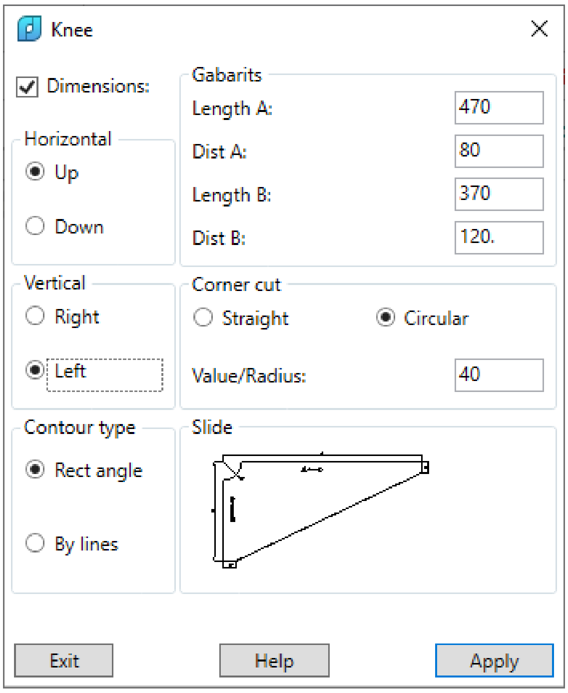
Checkbox **Dimensions** and radio buttons **Up**, **Down**, **Right** and **Left** ensure four kinds of dimensions location, for rectangle described in the fields **Length A** and **Width B** pf the area **Gabarits**.

Button **Apply** closes window and starts building rectangle.

Button **Exit** closes window with no actions. Button **Help** opens help topic.

Knee

Command **Knee** serves for creating parts having form of knee, with a straight angle between base sides or lying between two curved hull lines. Command opens dialog box shown on pic. 38.




Pic. 38. Window for parameters of knee

Knee parameters are entered in area **Gabarits**: **Length A**, **Length B**, **Dist A**, **Dist B**. In the intersection point for base sides there can be equal cut or scupper (type is defined by radio buttons **Straight** or **Circular**). Cut size is input in the field **Value/Radius** (for zero value no cut is built).

Areas **Horizontal** and **Vertical** define kind of dimensions location. They work only if checkbox **Dimensions** is set. Location kind is reflected on slide image and on position of base sides (illustrated by slide).

If checkbox **By lines** is activated then knee will be built using two reference objects (polylines, lines, arcs), without dimensions.

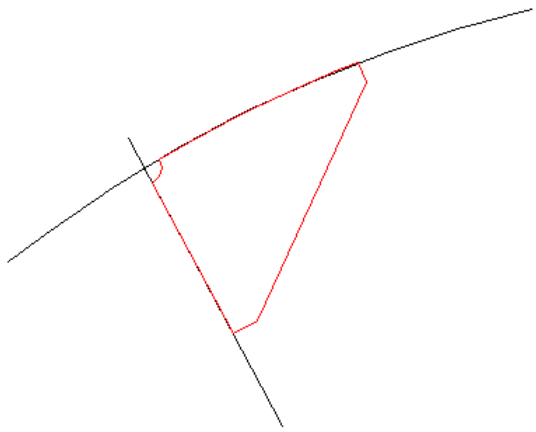
After selection of **By lines** clicking button **Apply** sends request into command line:

*Select reference line 1 <exit>:*

Select the first reference 2D line for knee. Second request:

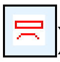
*Select reference line 2 <exit>:*

Select the second polyline. If polylines are not trimmed in the intersection point then they are to be picked in the side where knee must be built. Sample result of building knee part by lines is shown on pic. 39.



Pic. 39. Building knee by reference lines

Belt

Command **Belt** () uses upper web attachment line and calculates geometry of the developed upper-adjacent rectangular belt, with marking boundaries of plane areas. Belt thickness value and thickness direction are taken into account. Belt is an upper detail of welded stringer (T-beam, welded).

It is supposed that there is an on-screen attachment line created in WCS of the current drawing. Attachment line is not obligatory to be a single line in the drawing. It must be open 2D polyline, not fitted with spline. Line and arc are admitted too.

Command line receives the following requests:

*Building attached belt flange...*

*Attachment line:* select open attachment line (2D polyline, line or arc);

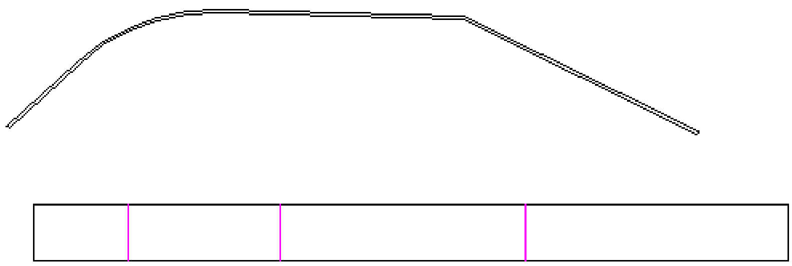
*Belt flange thickness:* enter belt thickness, mm;

*Belt flange width:* enter belt width, mm;

*Point defining thickness direction:* with mouse pick point to define the side where parallel line will be built on distance of thickness;

*Point for placing left down corner of a rectangular flange contour:* pick point that will become the left down corner of the belt part outer contour.

After entering data program converts line or arc to polyline (if selected not polyline), calculates belt length with considering location of neutral layer and builds full contour of longitudinal section for belt part, as well as rectangular contour of part development with marking boundaries of flat and arc areas of attachment line (pic. 40). Development dimensions are calculated with belt middle layer (in transverse section).



Pic. 40. Belt

Command TO KBAS

Command **To KBAS** () serves for moving existing objects (lines, arcs, circles, 2D polylines) to layer KBAS, with possibility of converting to polyline.

For splines, ellipses, 3D polylines and 2D polylines, fitted with spline, it is suggested approximation with polyline containing only linear segments, as well as projecting objects to XY plane of current UCS. Sample requests:

*Entities to be moved to the KBAS layer:*

*Select objects:* select spline objectн

*Objects in set: 1*

*Select objects:* Enter

*To replace by 2D projection to the XY plane of the current UCS? [Y/N] <Y>:* Y

*Object length (for reference): 698.7*

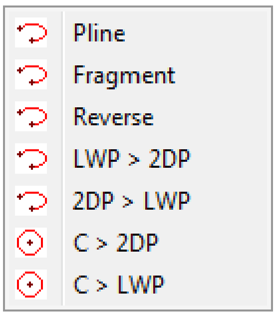
*Enter number of segments (more than 4) <27>:* 42

Instead of request on 2D projection for some objects there can be other question:

*To approximate in the current UCS? [Y/N] <Y>:* Y

Submenu Polylines

Submenu **PART > CONTOUR > POLYLINES** (pic. 40) contains some commands running operations on polyline objects that can be later used in building part contours.




Pic. 40. Submenu **POLYLINES**

Command **Pline** builds 2D-POLYLINE and immediately places it to layer KBAS.

Command **Fragment** allows selecting fragment from the existing closed or open polyline (2D-POLYLINE, LWPOLYLINE) and creates on the same layer new object with red color. Calculation is run with counting all the widths all the touched segments of the selected curve. Command requests:

*Creating polyline fragment.*

*1st point of fragment:*

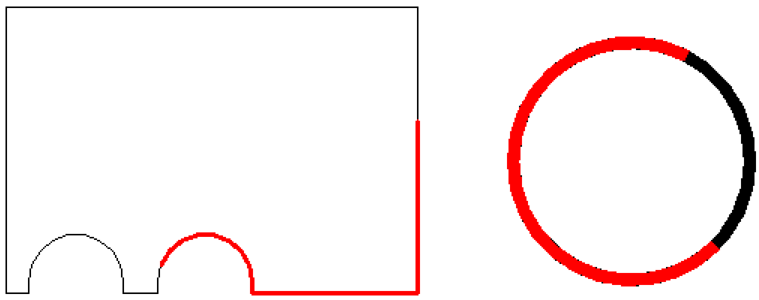
*2nd point of fragment:*

*Point inside selected fragment:*

*<Entity name: 21c1f030>*

**Note.** Resulting object has type **LWPOLYLINE**.

On pic. 42 there are two samples of building fragment (it has red color).



Pic. 42. Fragment creation

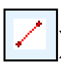
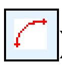
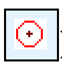
Both source polylines are closed, drawn with black color. Right polyline has nonzero global width. Intermediate points for selected fragments were picked so that fragment were created in the required part of contour (and there are always two segments for closed polyline). Program properly handles situation when the start polyline point appears inside fragment to be created.

Command **Reverse** serves for changing vertices sequence in polyline to an opposite one, for polylines of type 2D-POLYLINE, LWPOLYLINE.

Commands **LWP > 2DP** and **2DP > LWP** convert LWPOLYLINE to 2D-POLYLINE and vice versa.

Commands **C > 2DP** and **C > LWP** convert circle (entity of type CIRCLE) to 2D-POLYLINE and to LWPOLYLINE, preserving form.

### Commands Line, Arc, Circle

Commands **Line** (), **Arc** (), **Circle** () provides creation of lines (segments), arcs and circles on layer KBAS as outer contour elements.

**Note.** Circle is automatically converted to 2D-POLYLINE of four segments (1/4 of whole circle).

### Command Cut

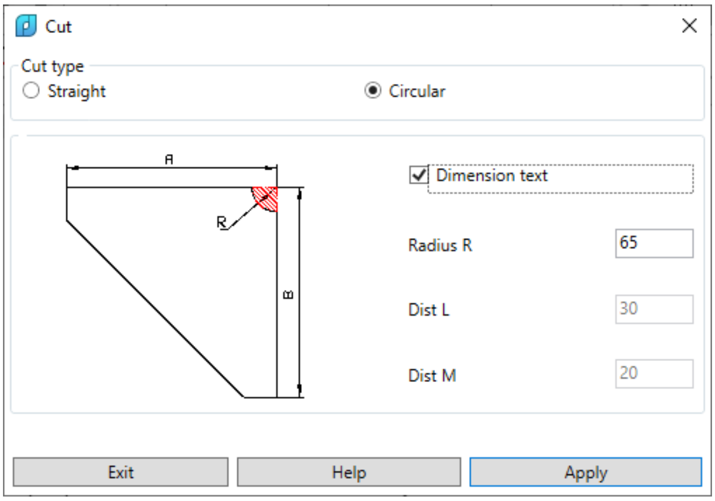
Command **Cut** is intended for trimming part outer contour corners by straight or radial cuts. Unlike commands for insertion of contour holes, command **Cut** does not save source outer contour on layer KBAS1, therefore in future operation of deleting cuts (as contour holes) and restoring previous line cannot be applied.

Command can be launched:

- with submenu item **PART > CONTOUR > Cut**,

- with button of toolbar **Contour_Outer**.

Command parameters are input by user in the dialog box **Cut** (pic. 43).



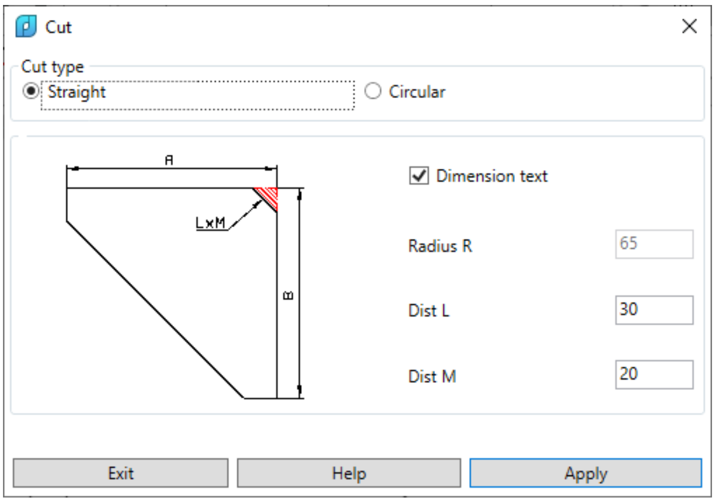

Pic. 43. Radial cut parameters

Radio buttons in area **Cut type** define cut configuration: by straight line (**Straight**) or по by radius (**Circular**). Checkbox **Dimension text** sets option of create dimensions and leader with text data of cut.

In the fields **Radius R** or **Dist L**, **Dist M** there are parameters being entered in different cut types.

Button **Apply** closes window with starting procedure of inserting cut in the interactive mode. Button **Exit** closes window with no actions. Button **Help** calls help topic.

On pic. 44 there is a window view for cut type **Straight**.




Pic. 44. Straight cut parameters

In the dialog box it is necessary to enter two distances from the corner for two edges.

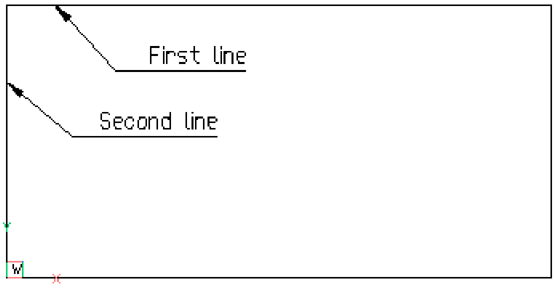
Field **Dist L** is used for cut length on first edge. Field **Dist M** is used for cut length on second edge.

After click on **Apply** there are requests (for straight cut):

*Select first line :*

*Select second line:*

On pic. 45 there are shown lines selected for straight cut.

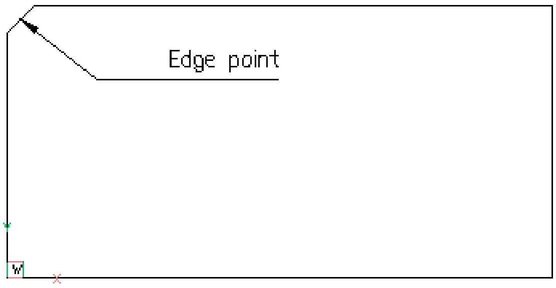



Pic. 45. Specifying lines for straight cut

If checkbox **Dimension text** was set then after creation of cut there will be next request:

*Specify point on chamfer edge:*

Specify point on cut edge to fix start point of leader with dimension text. On pic .46 edge point is shown with an arrow.

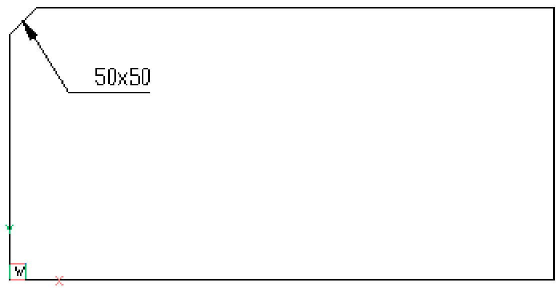



Pic. 46. Specifying edge point for leader start

Next request:

*Specify second leader point:*

Specify end point of the first leader segment. After that leader is created with parameters from dialog boxes on pic. 43 or 44. Next window **Cut** is opened again to continue applying cuts. To stop press **Exit**. On pic. 47 there is a sample result for straight cut (50x50).




Pic. 47. Final view after applying cut

All the cut settings are saved and suggested as default in the next session.

If command **Cut** is used with radius (to create scupper) then radius is entered in the field **Radius R** (pic. 43). Then press button **Apply**.

Sample requests of command **Cut** in the mode **Circular**:

*Active OSNAP is <End,Int>*

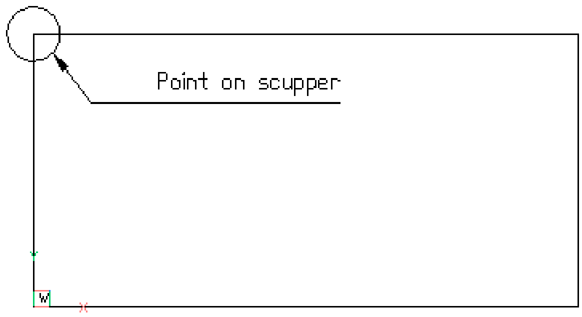
*Center of scupper R=50 <exit>:*

With this requests object snaps Endpoint and Intersection are activated.

It is necessary to specify point to become scupper center. Next drawing is being zoomed and shows red circle with the required radius. Command line gets message:

*Specify point on the scupper:*

It is necessary to pick point on that part of the circle that must become scupper (pic. 48).




Pic. 48. Specifying point for future scupper

If checkbox **Dimension text** is activated then after cut creation there will be request for radial dimension:

*Specify start point for dim text:*

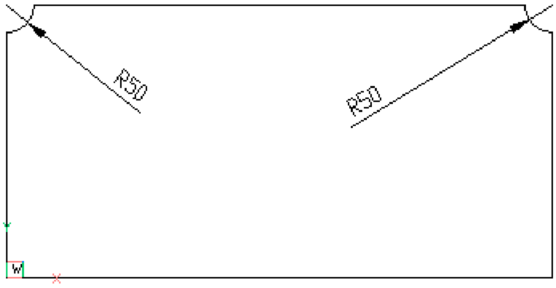
Select position of the future dimension text by left click. Program builds dimension with messages in command line:

*Active OSNAP is <End,Int>*

*Center of scupper R=50 <exit>:*

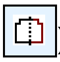
User may specify next scupper center point. If cut with the current radius is not needed the press Enter. Again dialog box **Cut** will be opened. To stop building cuts click button **Exit**.

On pic. 49 there is a sample results for two circular cuts with R=50.




Pic. 49. Part view with scuppers

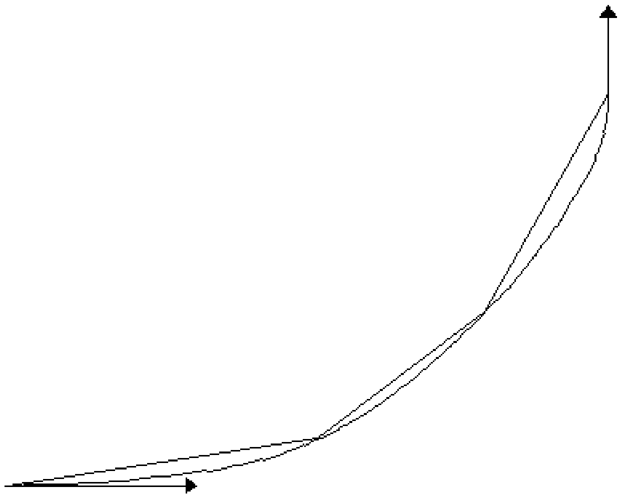
Mirror part

Command **MirrorPart** () serves for mirroring part located in the center of active drawing. As mirror axis there is used a vertical line going through the center point of the current part.

After full mirroring all the texts and dimension text are staying readable.

Ship line

Command **ShipLine** is used for quick creation of smooth curved line without knuckles, with no curvature breaks, without linear segments (only arcs). The line geometry is defined by a set of node points through which line is to go, with possible angular conditions (pic. 50).



Pic. 50. Ship line

**Note**. Other good results can be achieved using spline and its approximation by polyline (see submenu command **PART > CONTOUR > To KBAS**).

For two end points there can be set tangent angles. By automatical fitting position of node points is not changed.

Dialog of command **ShipLine**:

*Ship Line*

*Enter node coordinates:*

This request is repeated until user presses **Enter**, meaning end of points input.

*Enter value of MAX bend arrow for straightening <1>:*

MAX bend arrow must be defined for the case if created smooth curve (spline) will be replaced by arcs with radius greater than 100 meters. Next:

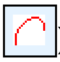
*Specify angles? [Both/Start/End] <no>:*

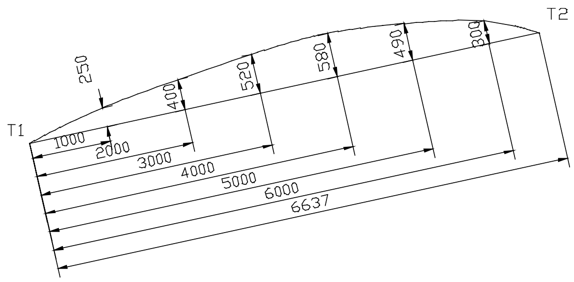
*Start angle (degrees):* if used options “Both” or “Start”.

*End angle (degrees):* if used options “Both” or “End".

After entering parameters smooth curve is being built (see pic. 50).

Sheet edge

Command **SheetEdge** () is intended to create a smooth line defined by chord end points and by a set of internal points specified by mouse or coordinates in the chord coordinate system (pic. 51). Resulting object is a polyline fitted by arcs.




Pic. 51. Sheet edge

Command **SheetEdge** dialog:

*SheetEDGE*

*Chord start point:*

*Chord end point:*

*Displacement on the chord and on the chord normal:* the request is repeated until user presses **Enter**.

On end of data input program builds a smooth line.

## 8 INSERTION OF PART HOLES

Forming part holes is done on layers: KHOLE - inner holes, KNOTCH - contour holes, KHOLEN - uncut holes (holes that are not cut while part manufacturing in the workshop).

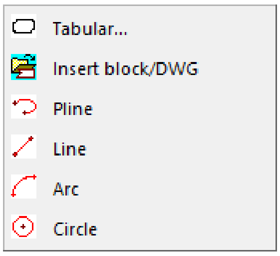
Holes of all the kinds can be created in three ways:

1) selecting tabular parametric hole;

2) building hole geometry with graphical editor tools;

3) inserting prepared DWG file with hole geometry from the folder *NSHIP\Dwg*.

For hole creation there is submenu **PART > HOLE** with embedded submenus **Inner**, **Contour**, **Uncut**. On pic. 52 there is a submenu **Inner** (submenus **Contour** and **Uncut** are identic).




Pic. 52. Submenu **Inner**

Command **Tabular** serves for insertion of tabular holes (their geometry is described inside software, but user is able to set parameter values and to save them as named sizes). Command **Insert block/DWG** inserts holes that are preliminarily built and saved in DWG files in the folder *NSHIP\Dwg*.

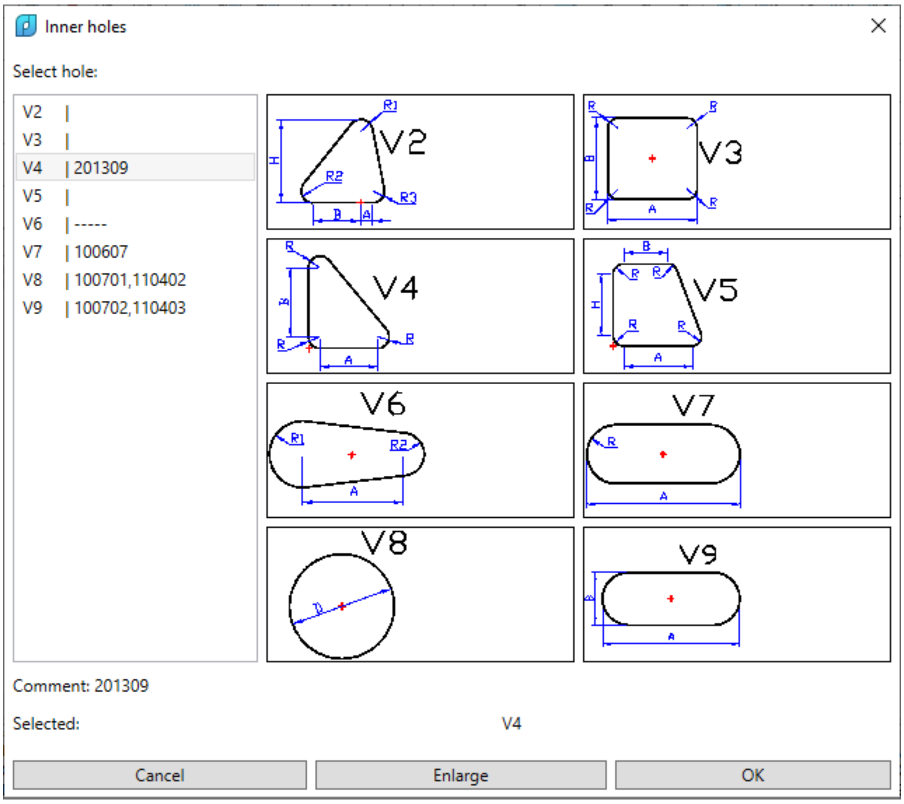
Commands **Pline**, **Line**, **Arc**, **Circle** are used for creating closed hole contours on the layer KHOLE with the tools of polylines, lines, arcs and circles. At the end user must assemble a single polyline for each hole. Otherwise program will try to assemble contours itself while saving part.

Contour holes are created with the similar submenu items but on the layer KNOTCH and open. Uncut hole contours may be either closed or open (layer KHOLEN).

Building holes is to be done **in WCS** that must be set as current UCS.

### Tabular inner holes

Command **HOLE > Inner > Tabular** opens dialog box of tabular holes (pic. 53).



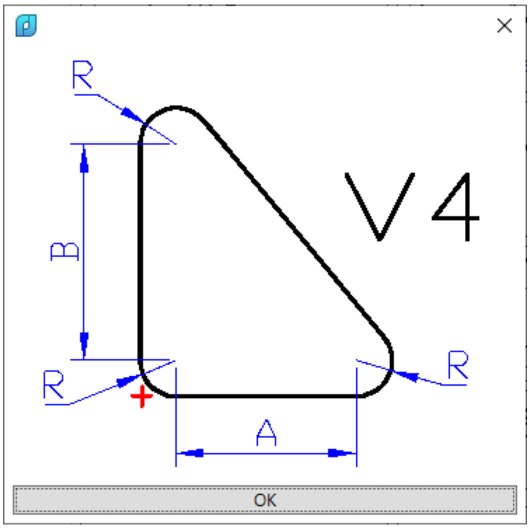

Pic. 53. Window for selection of inner hole type

In the left side of the window there is a list of lines with hole type names and comments attached to them. In the right side there are graphical images of hole types (slides).

Hole type can be chosen either by clicking slide in the right side or by marking hole line in the left side in the list **Select hole**. Name of the selected type is displayed in the lower side in the line **Selected**. Comment to type is shown in the line **Comment**.

Name of the inner hole type consists of symbol **V** and number. Each type has its own textual file of sizes in the folder *NSHIP\Tbl*. This file contains not only hole type sizes but comment too (alternate names and designations from the hole albums).

Button **Enlarge** allows to show increased hole type image to view geometry features (pic. 54).




Pic. 54. Slide with hole image

On the slide the insertion point is marked with red cross. Also there are displayed parameters defining hole size and geometry.

Button **Cancel** (see pic. 53) closes window with no actions. Button **OK** is intended for calling next window shown on pic. 55, for selection or entering hole parameters.

Dialog has seven areas:

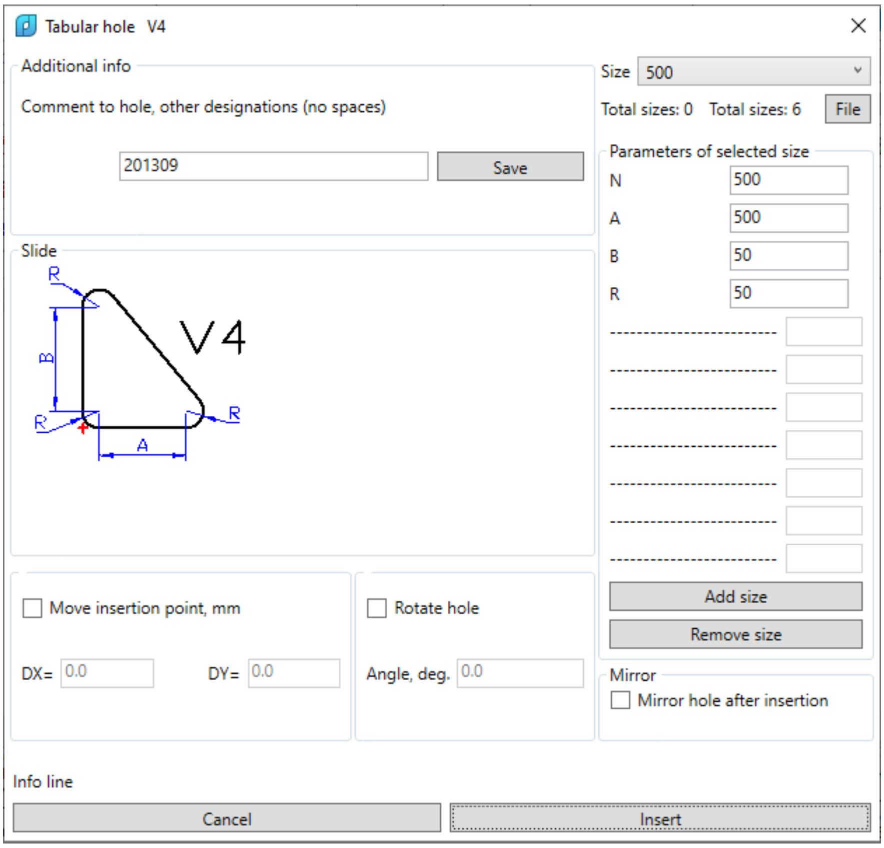
- **Additional info**, with current comment to the hole type (field **Comment to hole, other designations (no spaces)**) and button **Save** (for saving new comment);

- **Slide**, slide with hole and parameters image;

- **Move insertion point, mm**, movements **DX** and **DY** for moving hole after specifying insertion point (enabled only after setting checkbox);

- **Rotate hole**, hole rotation angle, in degrees (enabled only after setting checkbox);

- **Size**, area in the right side, contains name of selected size (drop-down list **Size**), Quantity of saved sizes for this hole type (**Total sizes**), button **File** for viewing sizes file, parameter value fields (**N**, **A**, **B**, **R**), button **Add size** and **Remove size** (for adding a new size and its parameter values and for deleting selected size);




Pic. 55. Window for setting parameter values for the hole type size

- **Mirror**, area with checkbox **Mirror hole after insertion**; if checkbox is set then after insertion hole is mirrored from right to left relative to vertical axis, drawn through insertion point;

- area in the lower side for program messages and with buttons **Cancel** and **Insert**.

Parameter values for the size selected in the list **Size** is displayed in the area **Parameters of selected size**. On dr. 51 hole V4 has four parameters: size name N, linear parameters A,B and radius R.

For adding new size it is necessary to enter in the upper line **N** name of the new size (it must not coincide with earlier saved sizes), to set its parameter values and to press button **Add size**.

For removing existing size it is necessary to select it in the list **Size** and click button **Remove size**. The only (last) size of any hole type cannot be deleted.

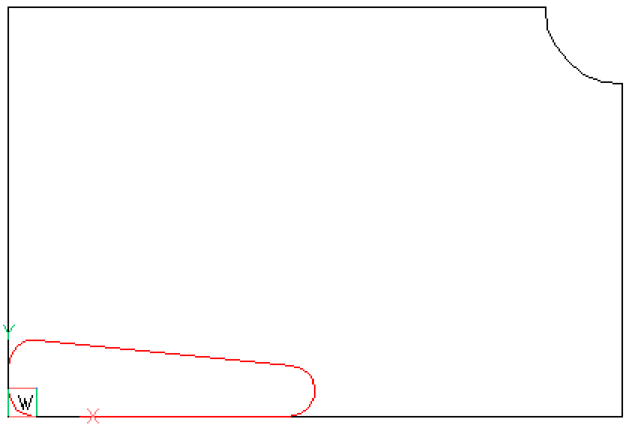
Procedure in inserting hole starts after clicking button **Insert** in the window shown on pic. 55. Window closes and command line receives text (in an example with hole V4):

*Created a sample of the hole V4 in the point (0,0), with red color.*

*Insertion points will be requested.*

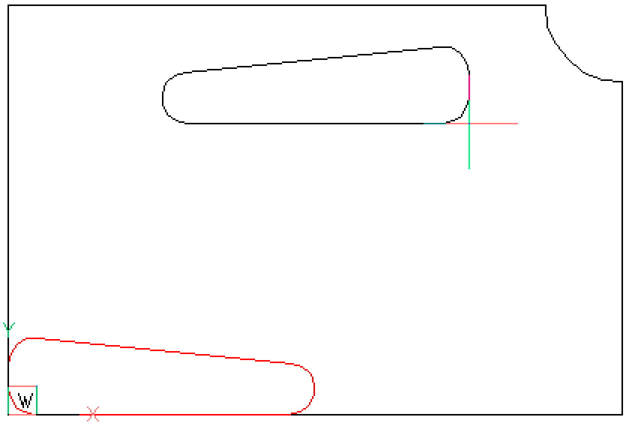
*Insertion point <exit>:*

In the point 0,0 there appears hole copy in red color (pic. 56).



Pic. 56. Inner hole copy in the point 0,0

If press **Enter** or right-click, then process ends without start. Therefore specify point placing cursor inside the part and left-clicking. On pic. 57 there is a first copy of inserted hole (in black color, already with values of movement, mirroring and rotation).

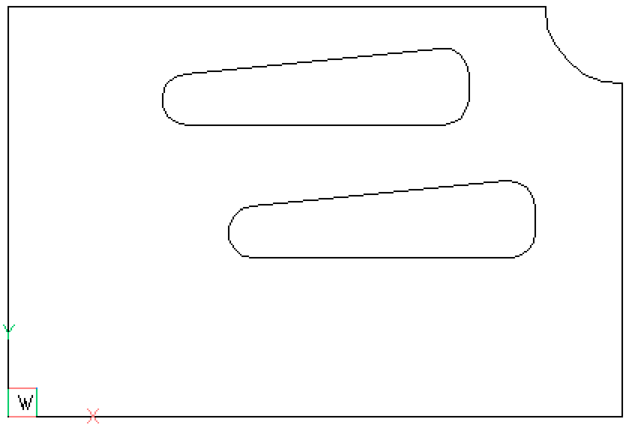


Pic. 57. First copy of inserted hole

New request is repeated:

*Next insertion point <exit>:*

Specify next insertion points and at the end right-click or press **Enter**. On pic. 58 there is a sample result of inserting two copies of hole V4 (with mirroring).

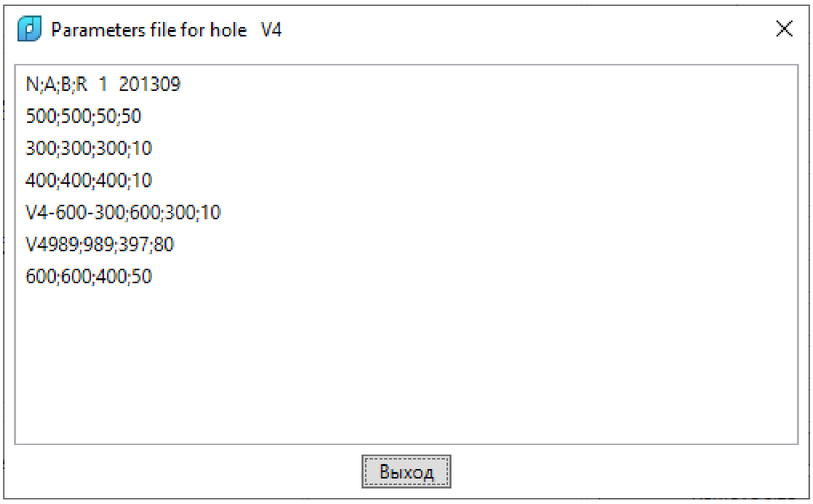


Pic. 58. Result of insertion

Tabular hole sizes file

Description of all the hole type sizes is saved in the file *NSHIP\Tbl\TV<type number>.<shipyard code>*. For example, for type V4 and shipyard with code TST file will be named *NSHIP\Tbl\TV4.TST*.

Viewing file is run with button **File** in the type size selection window (see pic. 55). Opens window of text editor. On pic. 59 there is a sample contents of such a file for hole type V4.




Pic. 59. Sample sizes window for hole type V4

In the upper line there are the following data:

- **N;A;B;R**, parameter names, specific for hole type (they are displayed in the window on pic. 55, spaces are not allowed);

- **0**, number of the last read size (count starts with 0);

- **201309**, comment text displayed in the area **Additional info** on pic. 55 and the field **Comment** on pic. 53 (spaces are not allowed).

Next there are lines with sizes data, in which parameter values are divided by semicolons (spaces are forbidden), e.g.:

- **500;500;50;50**, name **500**, A=500, B=50, R=50;

- **V4-600-300;600;300;10**, name **V4-600-300**, A=600, B=300, R=10.

It is recommended to use similar names inside one hole type. Manual editing size file is allowed (e.g. in Notepad) but with strict following file structure.

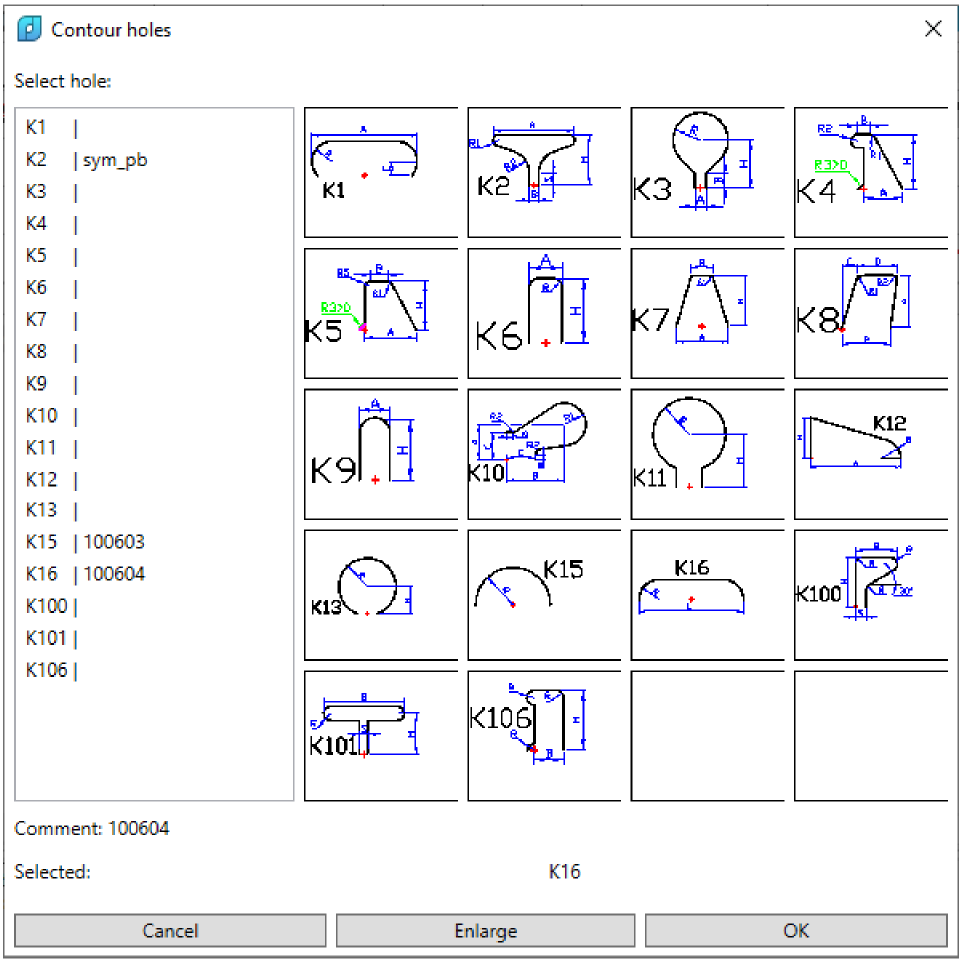
In the system installer there are included template files for all the tabular inner holes, shown on pic. 53 (from V2 to V9). Templates have the same names as size files, but with no extension: *TV2* (hole type V2), *TV3* (hole type V3), etc. In the first calculation contents of the template file is copied to the file with the same name but with extension containing shipyard code (e.g. *TV2.TST*). Further work is being done only with the file having extension.

**Note.** Shipyard code can be known from the order activation window (**BDATA > ORDERS > Activate order >** (select order) **> Code**) or in the window for viewing table *plants.dbf* (**BDATA > TABLES > AUXILIARY > Plants > Documentation code**).

### Tabular contour holes

Scheme of building tabular contour holes is similar to the scheme of tabular inner holes. Differences are in layer name (KNOTCH) and in open contour hole lines.

Command **HOLE > Contour > Tabular** calls dialog box for hole type selection (pic. 60).



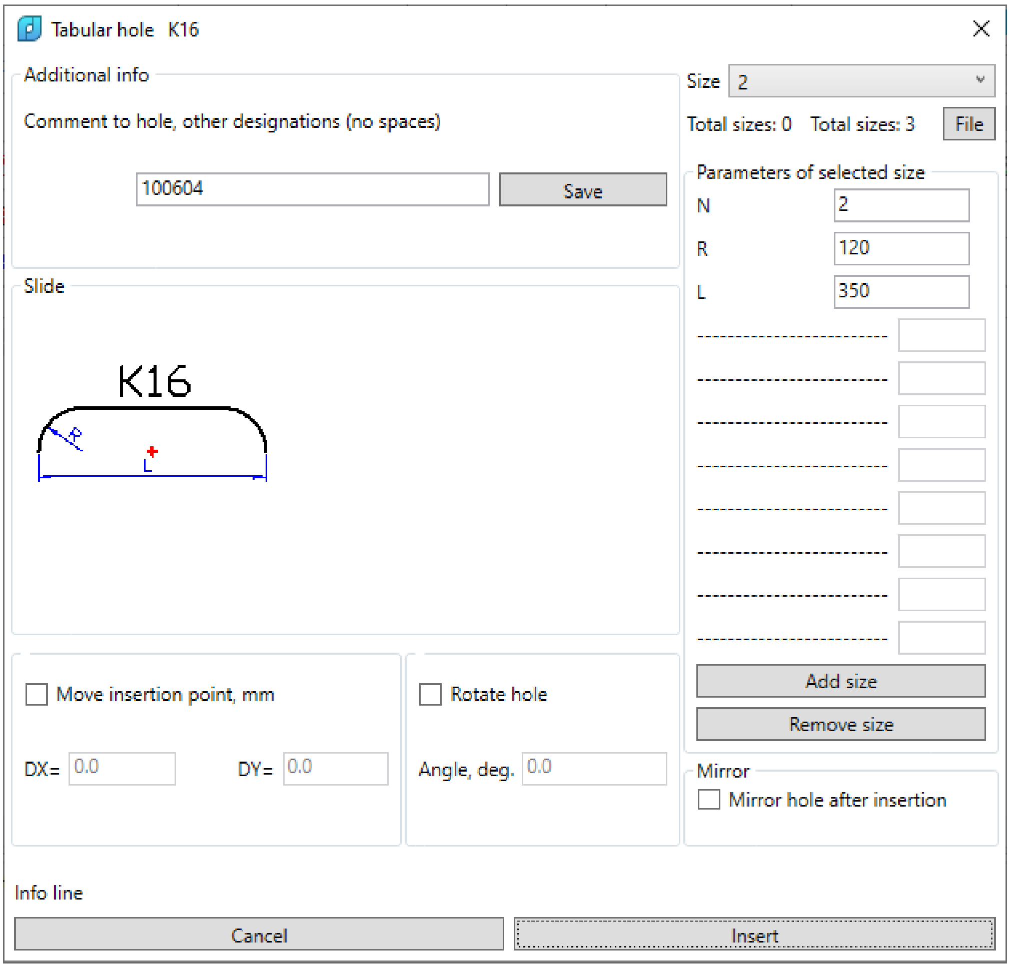

Pic. 60. Window **Contour holes**

In this window there are also displayed comments to contour hole types (like comments to inner hole types). After selecting type and clicking button **OK** user is redirected to the window of sizes and parameters (pic. 61).

Work with comments, parameters of movement, mirroring and rotation is similar to work in windows for tabular inner holes. New size can be added, existing size can be removed.

Size file names for contour holes are constructed by formula *NSHIP\Tbl\TK<type number>.<shipyard code >*.

Contour holes should be implemented into part outer contour. While saving part to DB program tries to extend hole lines up to outer contour if lines are too short or to trim if hole lines are too long.




Pic. 61. Window **Tabular hole K16**

After clicking button **Insert** in the dialog box from pic. 61 window closes and in the point 0,0 there appears a copy of hole in red color (pic. 62).


Pic. 62. Contour hole copy in the point 0,0

Messages are sent to command line:

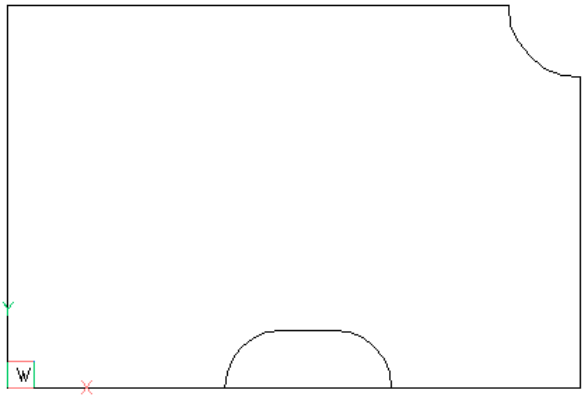
*Created a sample of the hole K16 in the point (0,0), with red color.*

*Insertion points will be requested.*

*NEArest snap is set on.*

*Insertion point <exit>:*

In this step all the copies of hole are inserted with the same values of movement, mirroring and rotation. Specify insertion point (points) and complete with **Enter**. Sample result is on pic. 63.



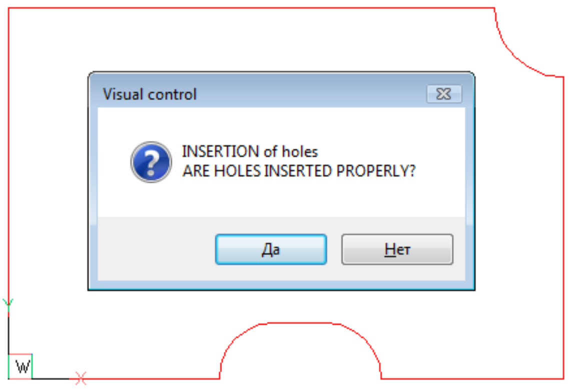

Pic. 63. Result of inserting contour hole

Run command for saving part (**PART > Save part**). Program starts procedure of attaching hole to outer contour, requesting:

*Pick point inside part*

Specify any point inside outer contour but outside hole(s).

Next message is optional and deals with visual control of hole attachment (pic. 64).




Pic. 64. Request on correctness of hole attachment

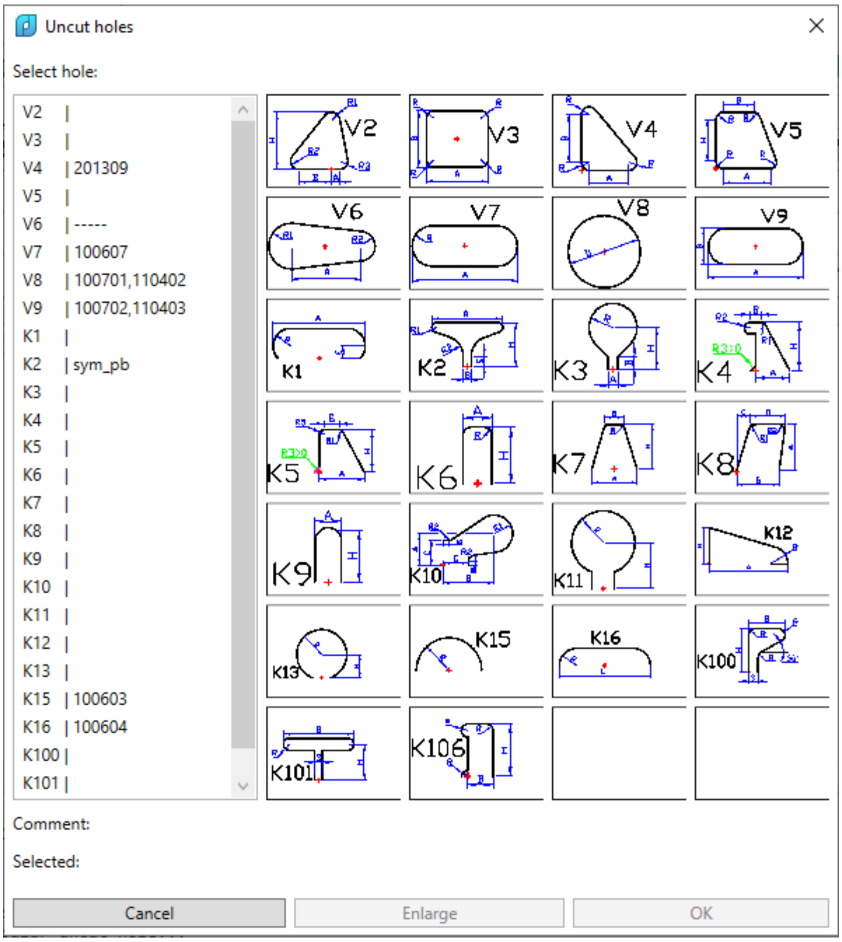
United contour is shown. If everything is OK, then click **Yes** (Да) and part will be saved.

If click **No** then part writing will be cancelled. In this case attachment of hole to the contour must be done manually by editing polylines of hole and outer contour. After that user must save part once more.

Tabular uncut holes

List of tabular uncut holes includes all the discussed tabular inner and contour holes. But lines of uncut holes are located on layer KHOLEN and they are ignored in the CNC programs for cutting parts.

Command **HOLE > Uncut > Tabular** calls dialog box of tabular holes (pic. 65).



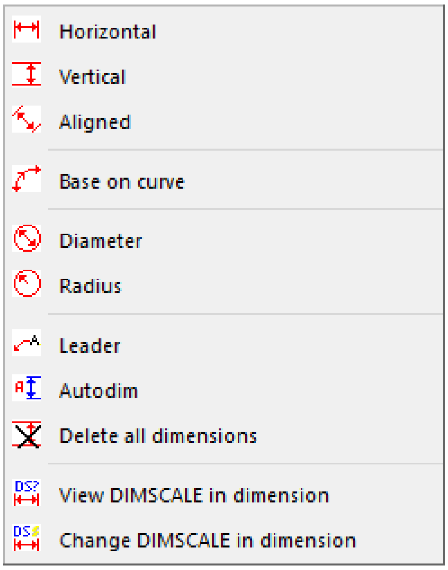

Pic. 65. Window **Uncut holes**

Hole type names start with symbols V and K. Process of their building is similar to the processes of insertion for corresponding inner and contour holes. They use the same files in folder *NSHIP\Tbl*.

During saving part with contour uncut holes, which names start with letter K, are not attached to the outer contour (no trimming and extending).

# 9 DIMENSIONS

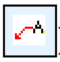
Part dimensioning is made with commands of submenu **PART > DIMENSIONS**, shown on pic. 66.

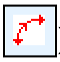



Pic. 66. Submenu **DIMENSIONS**

Command of any item in submenu **DIMENSIONS** seta layer DIM as current and shows on-screen part base contour, without allowances.

The following commands of submenu **DIMENSIONS** use dimensioning commands of graphical editor, but create dimensions on layer DIM: **Horizontal**, **Vertical**, **Aligned**, **Base on curve**, **Diameter**, **Radius**. They form specific dimension style and find corresponding scale.

Command **Leader** () builds its own leader without using editor command.

There is a specific opportunity to create dimensions with lengths measured along curved line, — e.g. for specifying distances along part edge from base point to holes. For this target command **Base on curve** () is applied. If lengths must be calculated with not counting inserted contour holes, then it is recommended to save preliminary curve without holes and before dimensioning to put it over the true edge of the part.

Dialog of command **Base on curve**:

*Pick start point of the edge part:* - specify start point of the fragment.

*Pick end point of the edge part:* - specify end point of the fragment.

*Select edge part for dimensioning:* - with mouse select inner point on one of the possible contour portions bounded by start and end points (if contour is closed there always be two portions). Contour fragment for dimensioning will be painted with red.

Next request asks for a point, that will fix equidistant location of dimension lines for all the chain of dimensions, not only between base and second points:

*Place dimension line:*

Next there is requested the first point (different from start point) on red fragment to be dimensioned (distance will be calculated along curve up to this point):

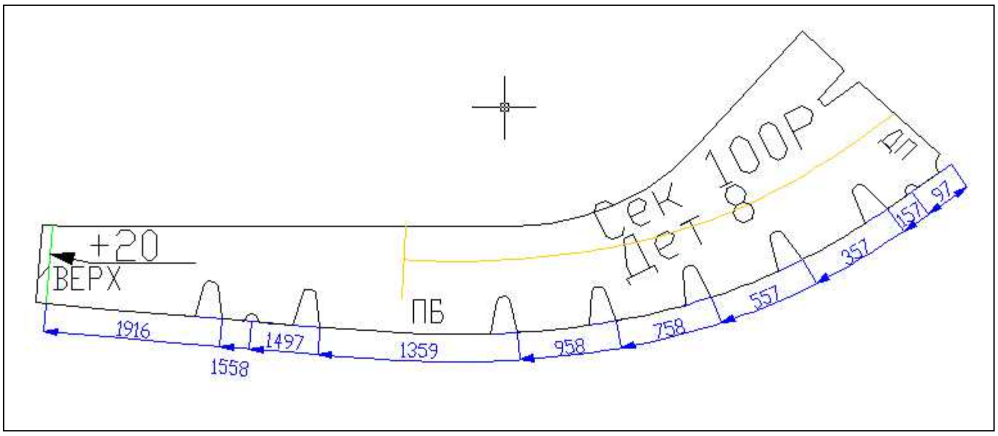
*Pick the 1 point up to which to measure distance from the beginning of the selected curve fragment:*

After that one by one there are requested next points on red curve (2, 3, 4, ...), up to which lengths to be measured. Command stops if user picks the end point of dimensioned fragment.

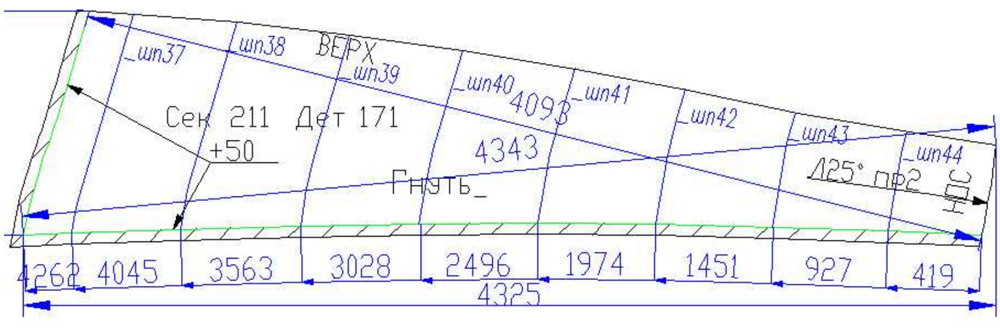
**Note**. Objects built by command **Base on curve** do not consist from dimension entities. This fact allows imitating curvilinear dimension lines.

There is a great variety of curved forms, so some dimension elements (e.g. arrows or dimension texts) may be placed improperly. In such a case some additional manual rework requires.

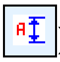
On pic. 67 there is a sample of dimensioning contour holes from base point, pic. 68 shows sample dimensioning of girders marks.

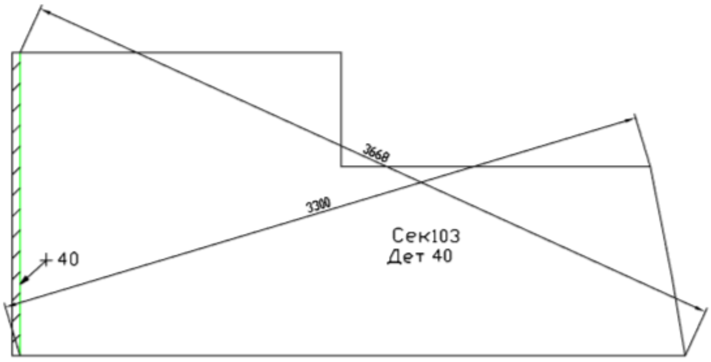



Pic. 67. Sample dimensioning lengths along curve up to holes

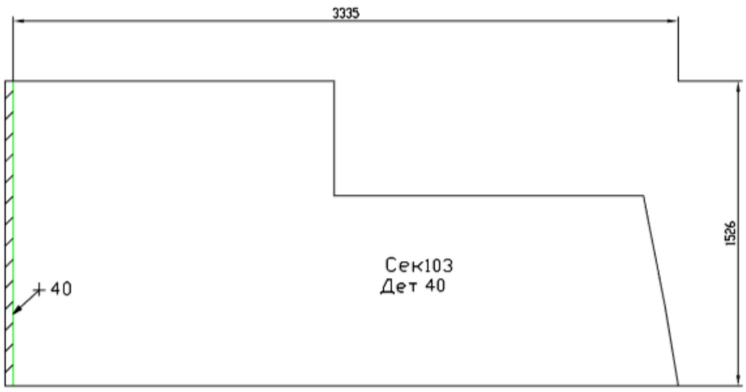



Pic. 68. Sample dimensioning lengths along curve up to girders

Command **Autodim** () allows in automatic mode to place on part diagonal dimensions (pic. 69), gabarit dimensions (pic. 70) with preliminary settings.

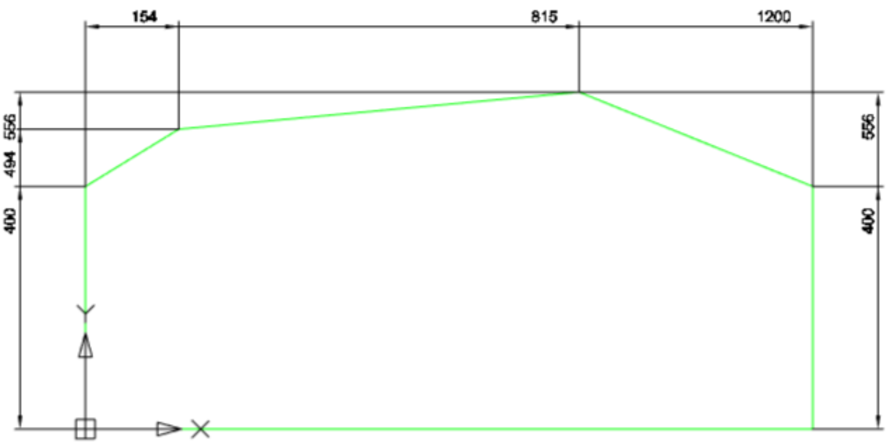



Pic. 69. Diagonal dimensions

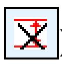


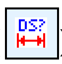
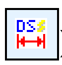

Pic. 70. Gabarit dimensions

On pic. 71 there is one more type of autodimensions: detailed.




Pic. 71. Detailed dimensions

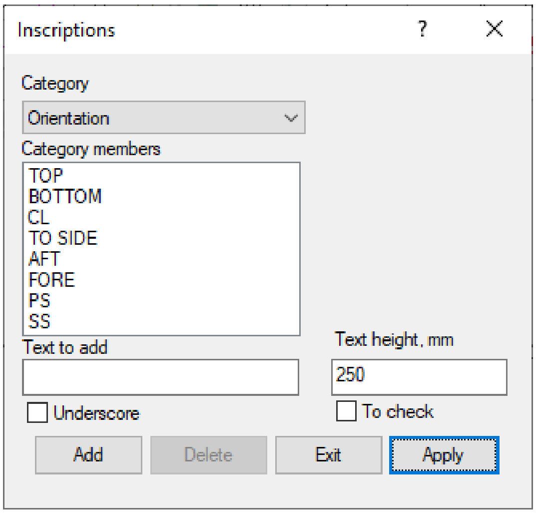
If it is necessary to delete all dimensions from layer DIM then user must run command **PART > DIMENSIONS > Delete all dimensions** ().

Two commands are designed for work with a global scale of dimension object (associated with the system variable DIMSCALE). Command **View DIMSCALE in dimension** () displays current value of decoration scale. Command **Change DIMSCALE in dimension** () modifies global scale to a new value providing better form of dimension object.

**Note**. Commands for work with a global scale do not influence on the objects created by the command **Base on curve** because these objects do not consist of dimension objects. This scheme enables imitation of curvilinear dimension lines.

## 10 ADDING TEXT INSCRIPTIONS

Writing text inscriptions (captions) inside part area is done with command **PART > Texts**. Command calls dialog box shown on pic. 72.

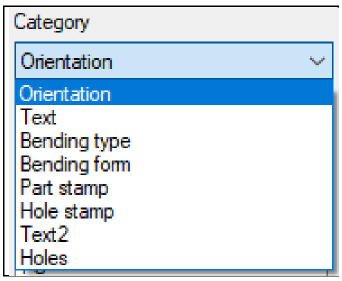



Pic. 72. Dialog box **Inscriptions**

Set of texts that can be added to the part sketch is stored in the file *StandardTechnoNoteList<lang>.ini*, from the folder with specific plant settings (e.g. *NSHIP\Plants_settings\<shd>*, where *<shd>* is a shipyard code, e.g. "TST"). <lang> is a 3 symbols suffix of current localization language ("_en" for English). Maximum number of lines in the file is 150. Inscriptions are divided into eight categories. List of elements in each category can be edited from the dialog. Default text height is calculated from the size of part outer contour. Selected inscription is inserted with button **Apply**.

Text categories

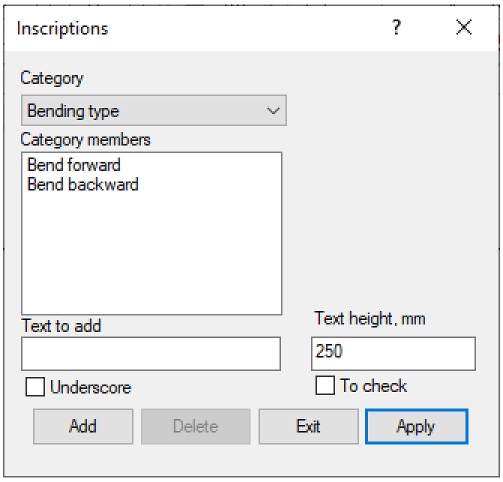
User must select a category (of 8) from the drop-down list **Category** (pic. 73).




Pic. 73. Drop-down list **Category**

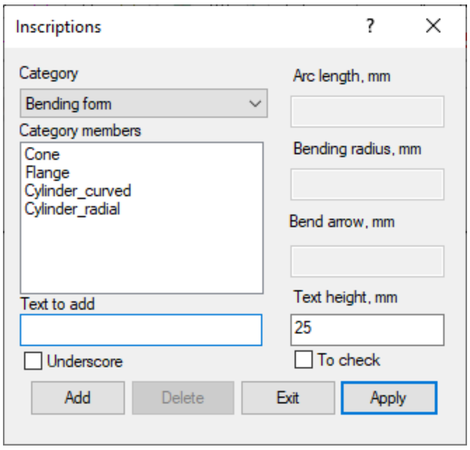
Category **Orientation** contains orientation texts (see pic. 72).

Category **Bending type** serves for creation of inscriptions connected to bending actions. On pic. 74 there is a dialog view for this category.



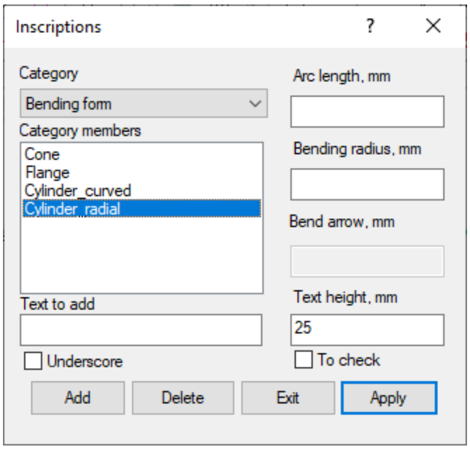

Pic. 74. List of inscriptions for category **Bending type**

Category **Bending form** is intended to define form for bending. Dialog for this category is shown on pic. 75.




Pic. 75. List of inscriptions for category **Bending form**

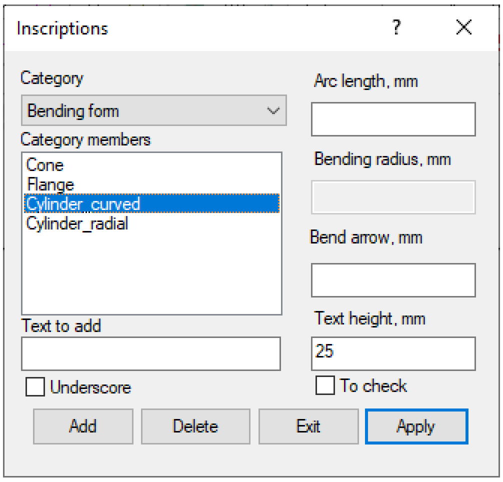
Page for this category has some additional features. If user selects element **Cylinder_radial** then window enables necessary fields (pic. 76).




Pic. 76. Dialog view for selection of element **Cylinder_radial**

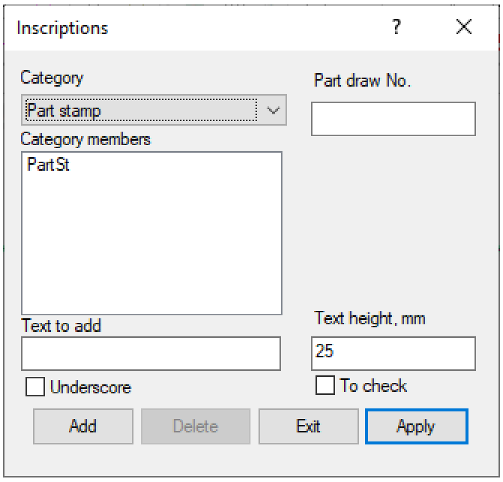
It is necessary to fill in parameters **Arc length** and **Bending radius**. These value must be integer and positive. If any of them is not filled command interrupts.

After selecting element **Cylinder_curved** (pic. 77) it is necessary to fill positive values for **Arc length** and **Bend arrow**.



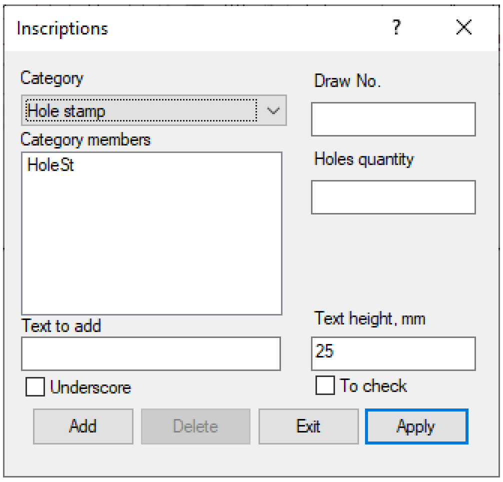

Pic. 77. Dialog view for selection of element **Cylinder_curved**

Category **Part stamp** (pic. 78) is intended to add designation of part stamp for part bending. This type if inscription requires filling parameter **Part draw No.**. Parameter value is textual (up to 5 symbols). Command is not run if parameter value is empty.



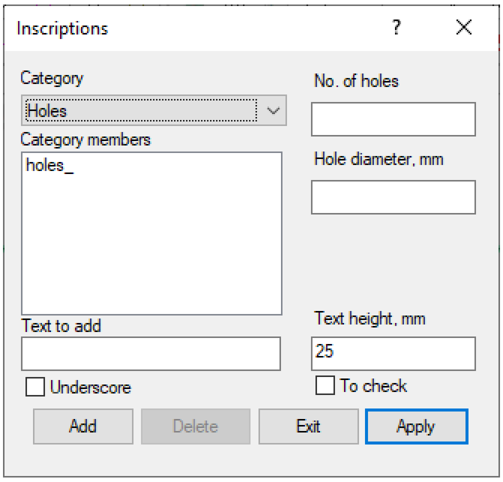

Pic. 78. Category **Part stamp**

Category **Hole stamp** is intended for adding designation of hole stamp. For adding this kind of inscription it is necessary to fill parameters **Draw No.** and **Holes quantity** (pic. 79). The first parameter is textual, the second one is a positive integer.




Pic. 79. Category **Hole stamp**

Category **Holes** (pic. 80) is used for texts concerning small inner holes and appends some information for creation of holes. Inscription requires parameters **No. of holes** and **Hole diameter, mm**. These values must be positive integer numbers.

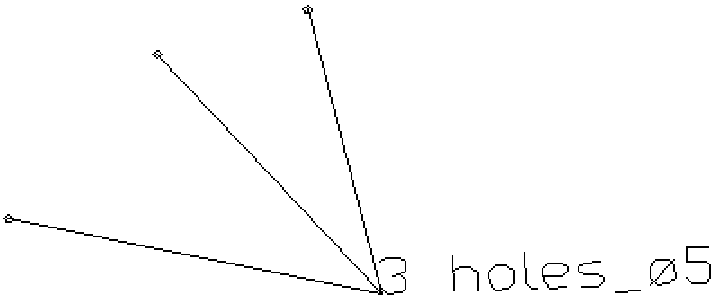



Pic. 80. Category **Holes**

After adding inscription program requests:

*To build? [Yes/No] <Yes>:*

In case of affirmative reply there will be built holes (with leaders), quantity of holes will be as defined in parameter **No. of holes** (pic. 81). Hole size will be taken from the parameter **Hole diameter, mm**. For positioning holes there will be issued requests *Circle insertion point:*.




Pic. 81. Holes with designation and leaders

After negative answer there will be no request for circle centers and leaders.

Categories **Text**, **Text2** can contain all the other inscriptions (contents is defined by user himself).

Everywhere adding inscription is run on clicking button **Apply** or by mouse double-click on category element. There are two standard requests:

*Point position:*

*Rotation angle:*

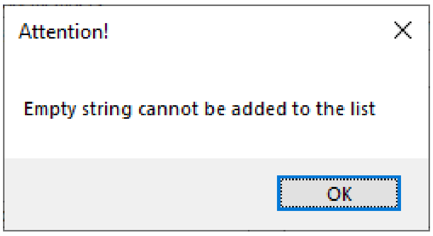
Height value is calculated automatically, as product of system variable DIMTXT and DIMSCALE values and is displayed in the field **Text height**.

**Note.** User must set an appropriate text height because default DIMSCALE value in nanoCAD is 100.

### Modification of text lists

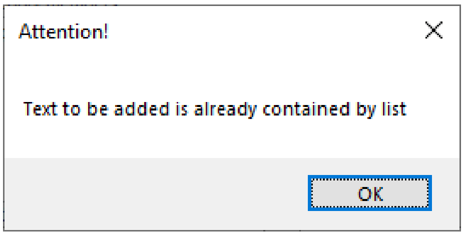
There are some tools for editing text lists by user. Button **Add** serves for adding text entered in the field **Text** to the list of current category.

If field **Text** is not filled but button **Add** was pressed then there will be message shown on pic. 82.




Pic. 82. Message about attempting to add empty string

In case if the added string already exists in the category then there will be issued a message shown on pic. 83.




Pic. 83. Message about attempting to repeat an element

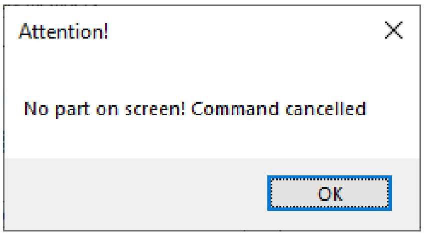
Button **Delete** is used for deleting selected string from the category.

Drop-down list **Category** is filled by strings from the file *StandardTechnoNoteList_en.ini* in the folder of current shipyard. Button **Exit** closes window.

### Possible errors

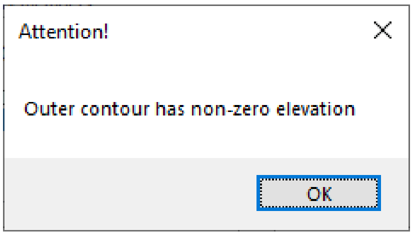
If checkbox **To check** is set then program tries to verify some errors in part before adding an inscription. If checkbox is cleared then verification is suppressed.

If outer contour on layer KBAS is missing then there is a message on pic. 84.



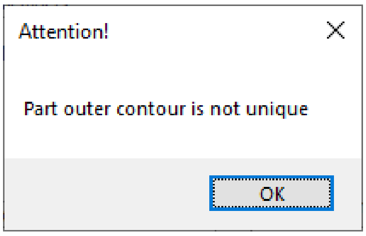

Pic. 84. Message about missing outer contour

If Z elevation of the outer contour is different from 0.0, then there is a message on pic. 85.



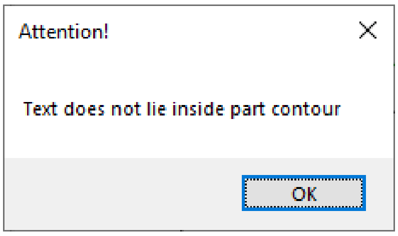

Pic. 85. Message about non-zero elevation

If there is more than one outer contour objects then a message is generated (pic. 86).




Pic. 86. Message after finding multiple outer contours

При непопадании надписи в поле детали выводится соответствующее сообщение (pic. 87).




Pic. 87. Message about text coming out of part contour

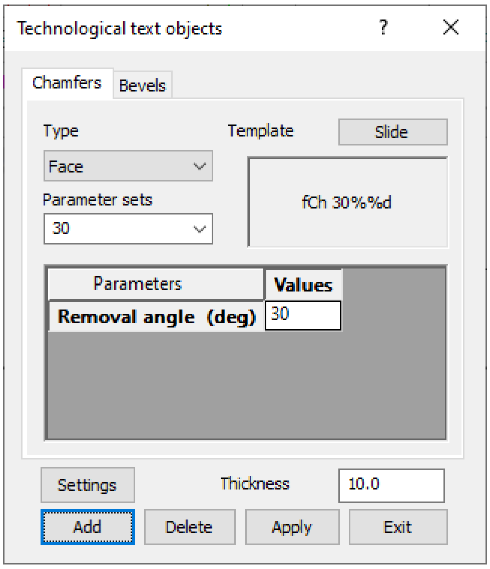
## 11 CHAMFERS AND BEVELS

Chamfers and bevels are technological operations for part edge, defined by welding conditions. Chamfers and bevels have parameters (removal angle, dulling etc.).

These operations require presence of special technological objects (multitexts with extended data) in part sketch. They are created with menu command **PART > TECHNOLOGY > Chamfers and bevels > Chamfer, bevel**.

### Chamfer by template

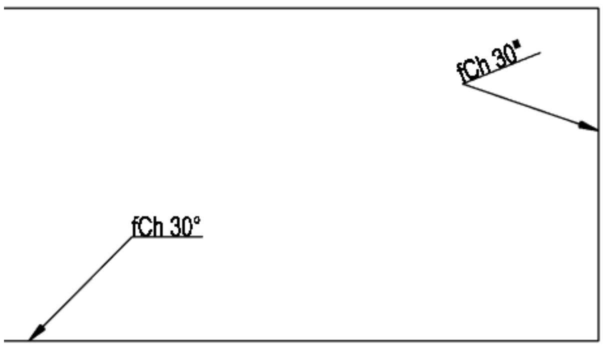
Creation of text object for chamfer is done with earlier created template. Command **Chamfer, bevel** opens dialog box with two tabs (pic. 88).




Pic. 88. Window for creation of technological texts, tab **Chamfers**

Tab **Chamfers** is used for creation of text object for chamfer (face, back, doublesided, symmetric, x-form). Chamfer object is an entity of type MTEXT, having specific extended data and located on layer FASKA. View of MTEXT object (one line, two lines, etc.) is defined by a template, and parameter values are set by user while inserting object. On pic. 88 there is shown a one line template, having form **fCh <Angle>%%d** (%%d is a degree symbol).

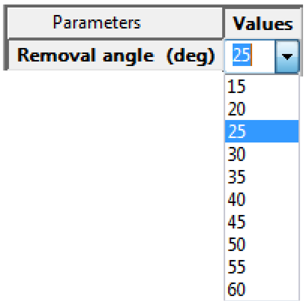
On pic. 89 there are sample chamfers.




Pic. 89. Sample chamfers

In the upper side of the window there is drop-down list **Type** for choosing chamfer type: **Face**, **Back**, **Doublesided**, **Symmetric** and **X-form**.

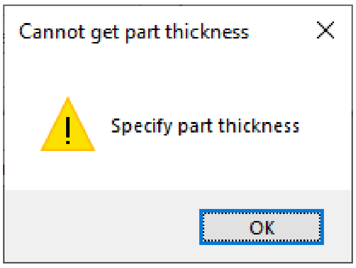
To the right side there is displayed view of the active template (some symbols are right-cut). In the center there is a table for setting parameter values for the current template. Table has two columns, for parameter names and for parameter values (removal, dulling, etc.). In the column **Values** there are implemented comboboxes with some set of standard values (pic. 90).




Pic. 90. Value list for parameter **Removal angle**

Sets of parameters can be applied further if saved. Used set of parameter values (for next sessions) can be saved with button **Add**. Saving is done into the file *Users\<work no.>\Account_en.xml* (inside current project_port folder) as a string. Future use is from the list **Parameter sets**. Selected parameter set can be removed with the button **Delete**.

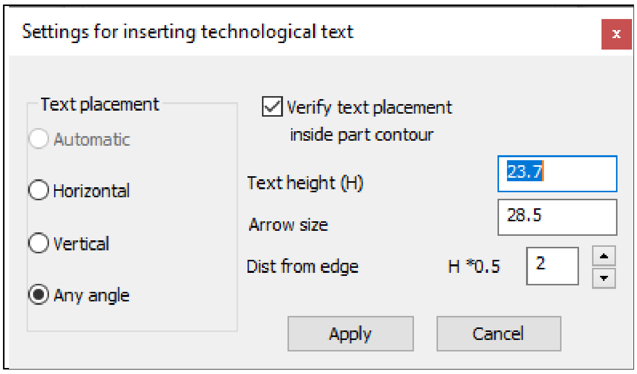
If the field **Thickness** has value 0.0 (this means that command **Open part** was not run), then after click on button **Apply** there will be an error message shown on pic. 91.




Pic. 91. Request for part thickness

For continuation user must enter value in the field **Thickness**, mm.

Button **Settings** (see pic. 88) opens window shown on pic. 92.




Pic. 92. Dialog box **Settings for inserting technological text**

In this window user can define placement view, verification of positioning inside part contour, text height, arrow size and distance from the part edge. All these data are calculated automatically, but here they can be redefined. If click **Apply** then settings will be saved.

After clicking button **Apply** in the dialog on pic. 88 procedure of text insertion starts and in coordination with the text template. There are requested the following data:

– leader start point (on part edge);

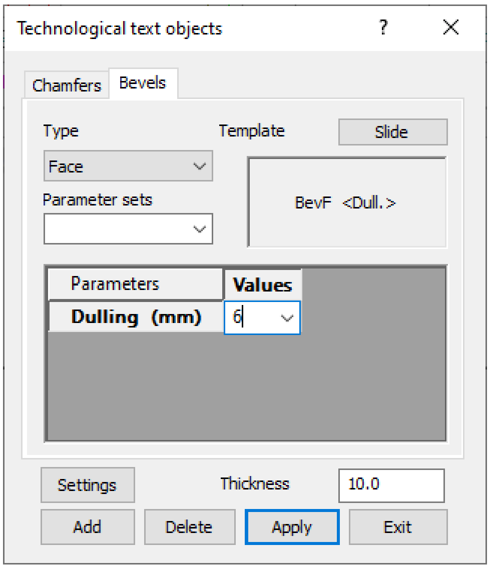
– text insertion point;
– text rotation angle.

**Note.** Setting checkbox **Leaders for technological texts** (see pic. 6) provides building leader line to chamfer and bevel text.

Leader must start on the part outer contour, to which chamfer will be applied in the workshop. Settings can define text inclination angle, including parallel to axis OX or to axis OY. If it is not done then orientation angle will be requested after insertion point.

By default settings confirm verification of technological text placement. Text position is checked for being completely inside the part contour. There are also will be checked position of the second leader point, second line point and two upper points of mtext. If norming mode is set on then user will be requested for two points on the part outer contour (for application of chamfer operation).

On pic. 93 There is an illustration to the process of entering parameter values for face bevel (uneven analog of chamfer but not so sharp).




Pic. 93. Dialog box for technological texts, tab **Bevels**

Tab **Bevels** is used for texts denoting bevel operation (one-sided, face or back, and double-sided).

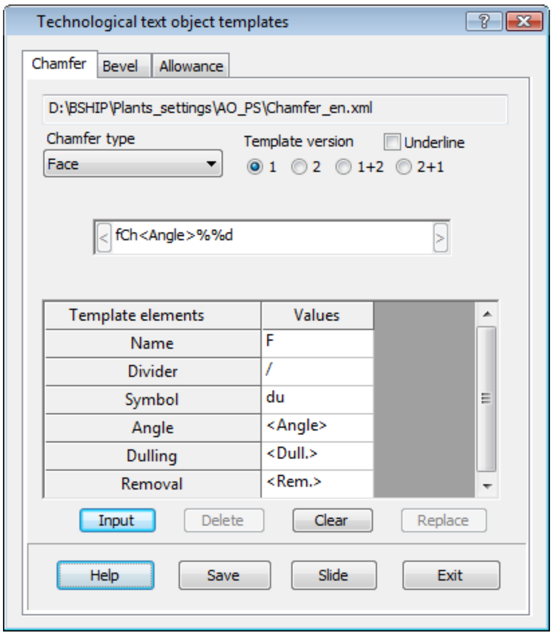
Work with **Bevels** tab is similar to work with **Chamfers** tab.

Command **PART > TECHNOLOGY > Chamfers and bevels > Show edge with chamfer** allows to show in temporary red color the portion of the contour where chamfer/bevel was attached. Command can be run only if norming mode was set, because in this mode attachment points coordinates are saved in the contour extended data.

Command **PART > TECHNOLOGY > Chamfers and bevels > Delete chamfer, bevel** removes chamfer or bevel data from the part.

Template for chamfer, bevel

Template for chamfer/bevel form is very important. For its settings there is dialog box **Technological text object templates** called from the window **Set part attributes** (see pic. 4), after clicking button **Technology** and in the next window (see pic. 9) after clicking button **Chamfers, bevels, allowances**. Dialog box is shown on pic. 94.




Pic. 94. Dialog box for tuning templates of chamfers, bevels and allowances

System deals with three objects (multitexts), each of them has several types. There are four types for chamfer (face, back, doublesided, symmetric and X-form), two types for bevel (face and back), two types for allowance (assembly and bending). Settings for all types of one object are saved in one template XML-file.

Here are names of template files:

- *Chamfer_en.xml* for chamfers,
- *Lask_en.xml* for bevels,
- *Allowance_en.xml* for allowances.

Template files are located in the folder *NSHIP\Plants_settings\<shd>*, where *<shd>* is folder name for the shipyard settings (saved in plants.dbf, for example *AO_PS*).

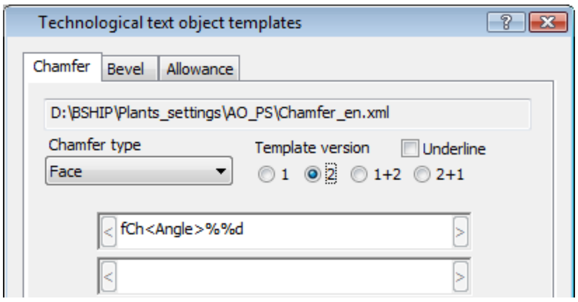
Next is some discussion for setting process on example of chamfers (tab **Chamfer**).

Drop-down list **Chamfer type** is intended for selection of chamfer type to be tuned. Elements in it: **Face**, **Back**, **Doublesided**, **Symmetric** and **X-form**.

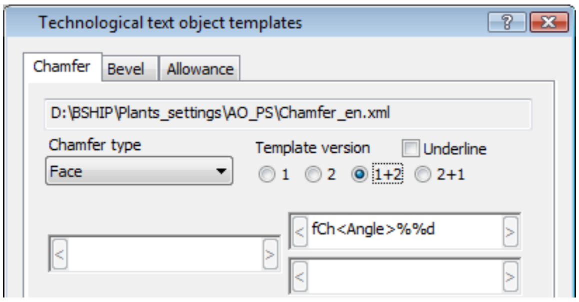
Group of radio buttons **Template version** defines view kind of template for MTEXT used as chamfer object:

- **1** - one line;
- **2** - two lines, without additional string to the left or to the right;
- **1+2** - two lines, with additional string to the left side;
- **2+1** - two lines, with additional string to the right side.

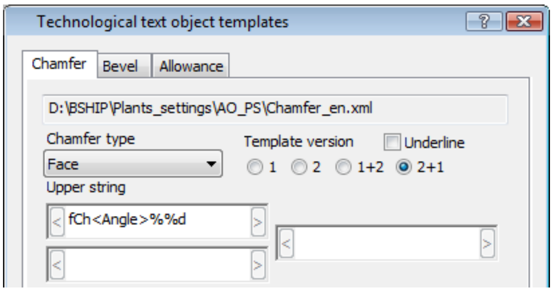
On pic. 94 there is shown the simplest option, one line (with one parameter). Choosing other radio button changes window view, illustrating external view of MTEXT for future chamfer object (pic. 95-97).




Pic. 95. Option **2**




Pic. 96. Option **1+2**




Pic. 97. Option **2+1**

Each template component is shown in the form of a string bounded by signs < and >. Editing for each component is done individually. If template will consist of two or three pieces then before editing it is necessary to left-click inside the corresponding component. Program reacts with additional text under the types list (on pic. 97 it is **Upper string**, i.e. upper tile of the template is being set). Other kinds of additional text are **Left string**, **Lower string**, **Right string**.

Checkbox **Underline** serves for building additional line like underscore of the upper string.

**Note.** Creation of leader to the chamfer text depends on general setting **Leaders to technological texts** in the dialog box **Request management** (see pic. 6).

Any chamfer string (line) is assembled from the allowed symbols: **Name**, **Divider**, **Symbol**, **Angle**, **Dulling**, **Removal** (pic. 98).

| Template elements | Values |
|---|---|
| Name | F |
| Divider | / |
| Symbol | du |
| Angle | <Angle> |
| Dulling | <Dull.> |
| Removal | <Rem.> |

Pic. 98. List of usable parameters

User must include into the template only symbols used by current shipyard for current project. Explanations:

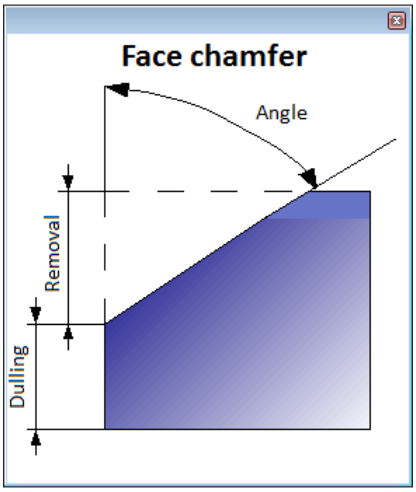
- **Name**, textual designation (name) of chamfer (e.g. **CH**, **F**, **Chbck** etc.);
- **Divider**, dividing symbol between parameters (standard ones are space, /, x, etc.);
- **Symbol**, text inserted before number (e.g. **du**);
- **Angle**, parameter value for chamfer angle in degrees (after insertion **<Angle>** will be replaced by number);
- **Dulling**, parameter value for dulling in mm (after insertion **<Dull.>** will be replaced by number);
- **Removal**, removal value in mm (after insertion **<Rem.>** will be replaced by number).

For the first three table elements (**Name**, **Divider**, **Symbol**) there is suggested some set of standard values, but user may add his own value.

To transfer element into string of the template it is necessary to select value in corresponding cell of the column **Values** and click button **Input**. If some element is added by error, then user must left-click on it in template (element will be highlighted with blue) and then click button **Delete**. To insert a missing element it is necessary to select element (it will be highlighted with blue) after which there must be inserted missing symbol, then select new element in column **Values** and press **Input**.

After finishing template settings it must be saved with button **Save**.

By button **Slide** user can view illustration to parameters that can be applied to the template (pic. 99).



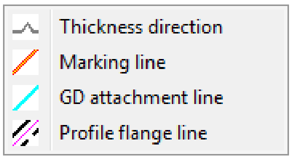

Pic. 99. Face chamfer parameters

Similar settings are done with templates of other chamfer type (back, etc.).

Bevel settings are very much alike, but with less number of types.

## 12 TECHNOLOGICAL LINES

Adding technological lines and connected texts is done with submenu **PART > TECHNOLOGY > Lines**, shown on pic. 100, or with toolbar **Technological lines**.




Pic. 100. Submenu **Lines**

Command **Thickness direction** ( ) calls procedure of forming either two entities of type LINE or angle sign inserted into marking line (sign case is defined in settings of module **Part**), on layer RAZM – for denoting thicknesses of nearby structures.

Command dialog for the first case:

*Specify point on path rib:*

*Rotation angle:*

Command dialog for the second case:

*Size of triangle side for direction thickness<56>:40*

*Specify point on path rib:*

*Specify direction of thickness:*

Command **Marking line** ( ) runs operation of moving marking line to the layer RAZM.

Command dialog:

*Select marking lines:*

*Select objects: found: 1*

*Select objects:*

Command **GD attachment line** ( ) calls procedure of moving rib trace lines to layer SLED and setting proper linetype.

Command dialog:

*Girder trace lines [Face/Back] <F>:*

*Select path of rib:*

Command **Profile flange line** ( ) serves for changing lines layer to MARK and setting linetype (visible/hidden).

Command dialog:

*Linetype for profile flange [L face/V back] <L>:*

*Select profile flange lines for moving to MARK layer:*

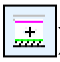
*Select objects*

## 13 ALLOWANCES

Command **PART > TECHNOLOGY > Allowance > Build** and button  of the toolbar **Technology** serve for generation of allowance segments on outer contour and inner hole contours. Part contour must be closed. Allowance size is entered as number in mm, that can be negative too. On outer contour and inner holes allowance is built to proper side (outer contour area increases, inner hole area reduces).

Allowance designation is a specific technological text, with possible leader. Created as block reference with extended data, used later in forming part handling rout in the workshop. If mode **Create data for norming and workshop handling** (see pic. 9) is not set, then extended data will be not attached to the contour.

Text external view is defined by template created in the window on pic. 94 (tab **Allowance**).

**Note.** Bending allowances are built with command **PART > BEND > Bending allowance** ().

Creation

Boundaries of the fragment to which allowance should be applied, is defined by two points on contour. The third point selects edge of application, from two possible. The third point serves as start leader point. To set allowance to all the contour it is necessary to the pick the same point as the first and as the second (it is convenient to use combination “@0,0”). In the last case The third point will be not requested. The same contour can have several allowances, even “allowance to allowance”.

Dialog of command **Allowance**:

*Allowance value:*

*Specify the 1st point on the contour:*

*Specify the 2nd point on the contour:*

*Select edge:*

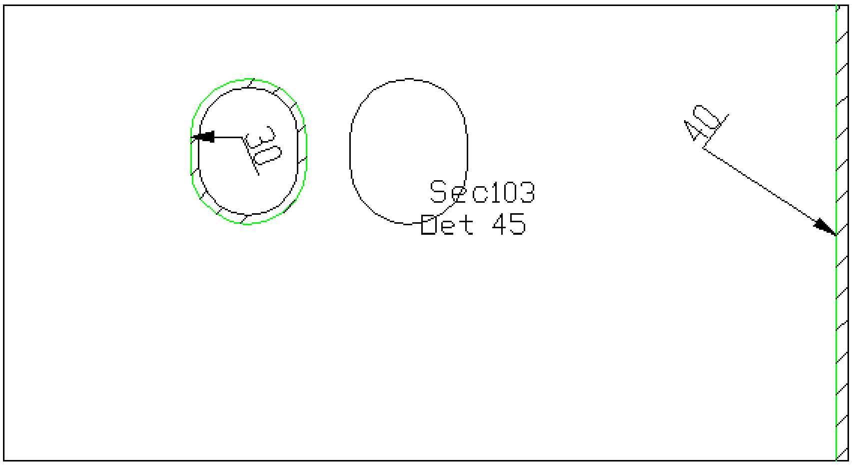
*Insertion point:*

*Text rotation angle:*

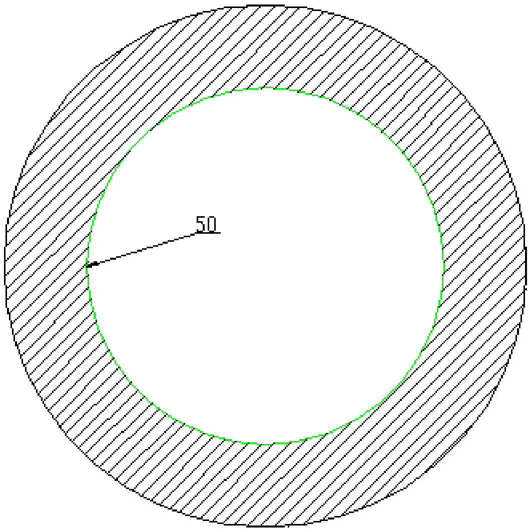
*Select next edge if you need an additional leader with the same allowance or press <Enter>:*

Allowance area is hatched with template ANSI31. Hatch lines inclination is 45 degrees by default, but can be changed in the attributes settings window (see pic. 4).

Sample results of building allowance to the outer contour and to inner hole are shown on pictures 101, 102.




Pic. 101. Sample of allowances to part contour fragment and to inner hole



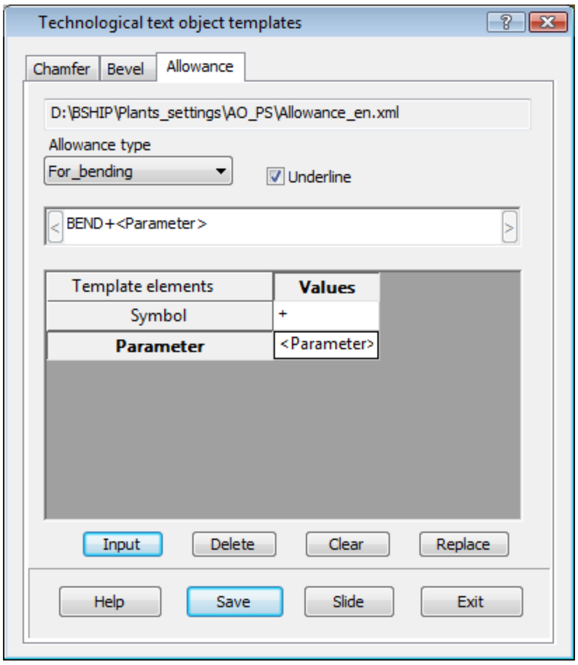

Pic. 102. Sample of allowance to the whole part contour

Command **PART > TECHNOLOGY > Allowance > Remove** ( ) is created for removing earlier built allowances and restoring source contour.

Template for allowance

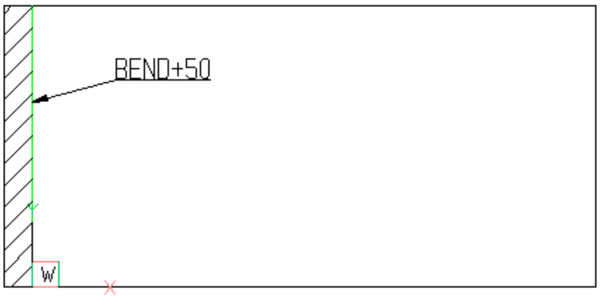
Tuning is done on the tab **Allowance** in the window **Technological text object templates** opened with **PART > Settings > Technology > Chamfers, bevels, allowances** (pic. 103).

There are two allowance types which form can be set. They are in the drop-down list **Allowance type**: **For_assembly**, **For_bending**. Checkbox **Underline** manages adding a line, like underscore line.




Pic. 103. Tab **Allowance**

As for chamfer, view of allowance text object is tuned in the field bounded with signs < and >. Only one line template is allowed. Constructing template is run with the help of elements **Symbol** and **parameter**. As standard values of **Symbol** there are offered values **BEND**, **All.** and **+**, but user may enter his own text value. On pic. 103 there is shown a template in the form **BEND+<Parameter>**. After insertion block attribute text changed to **BEND+50** (pic. 104).

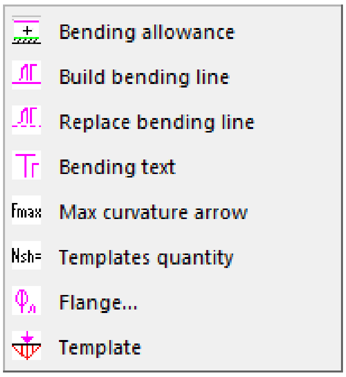



Pic. 104. Bending allowance

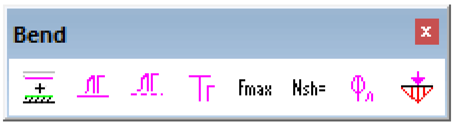
Template is saved in the file *Allowance_en.xml* in the folder *NSHIP\Plants_settings\<Shipyard>*, where *<Shipyard>* is a folder name for current shipyard technological settings.

## 14 BENDING OBJECTS

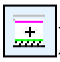
Creating bending technological objects (lines, texts) is done with commands of submenu **PART > BEND**, shown on pic. 105, and of toolbar **Bend** (pic. 106).

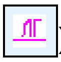



Pic. 105. Submenu **BEND**




Pic. 106. Toolbar **Bend**

Command **Bending allowance** () serves for applying bending allowance in the manner similar to command **Allowance** discussed earlier. The main difference is a text defined in the allowance template (for example: *All.40 for bending*). Template contents is tuned in the settings window (see pic. 103).

Command **Build bending line** () is used for drawing straight bending line defined by two end points. Bending line is to be placed on layer TIPDET.

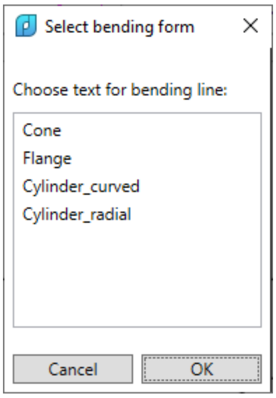
Command dialog:

*Bend to face [Yes/No]? <Y>:Y*

*Specify start of bending line:*

*Specify end of bending line:*

Next there is opened window for text selection that will be placed on bending line (pic. 107).




Pic. 107. Window for selection of text to bending line

List with bending form texts is extracted from the general inscriptions file *StandardTechnoNoteList_en.ini* (category **Bending form**) of current shipyard from folder

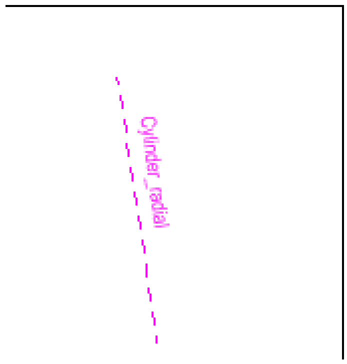
*NSHIP\Plants_settings\<Shipyard>*. User can edit this file himself manually or with command **PART > Texts >** category **Bending form**.

After text selection program requests:

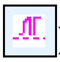
*Insertion point:* - pick point.

*Rotation angle:* - pick two points defining inclination of text.

Sample line is on pic. 108.




Pic. 108. Sample bending line with text

Command **Replace bending line** () changes object's layer to TIPDET and linetype to CONTINUOUS or DASHED1, depending on bending direction. Moreover, command allows to change bending text contents.

Command dialog:

*Change bending direction? [Yes/No] <N>: Y*

*Bend to face [Yes/No]? <Y>: Y*

*Select bending lines <exit>:*

*Select objects:*

For finishing selection press **Enter** or use mouse right-click.

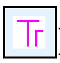
*Change bending texts? [Yes/No] <Y>: Y*

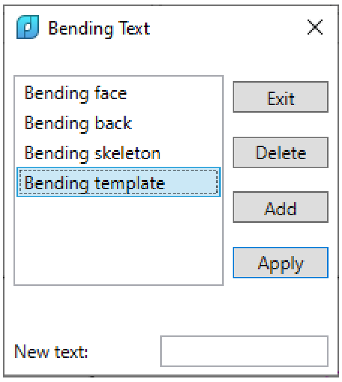
*Select single line text (TEXT entity) <exit>:*

Select text object to be modified. Object will be deleted and window for new text selection will be opened (see dr. 100). Select text string and click **OK**. Insertion point and rotation angle are requested and text will be added to the part drawing. Next the request is repeated:

*Select single line text (TEXT entity) <exit>:*

Pressing **Enter** stops command of changing bending objects.

Command **PART > BAND > Bending text** () allows to insert any prepared texts for different bending forms. Such texts are saved in the file *NSHIP\Tbl\Tgibka_en.<shd>* (*<shd>* is a shipyard code, e.g. TST). Command opens dialog box **Bending Text** (pic. 109).




Pic. 109. Window **Bending Text**

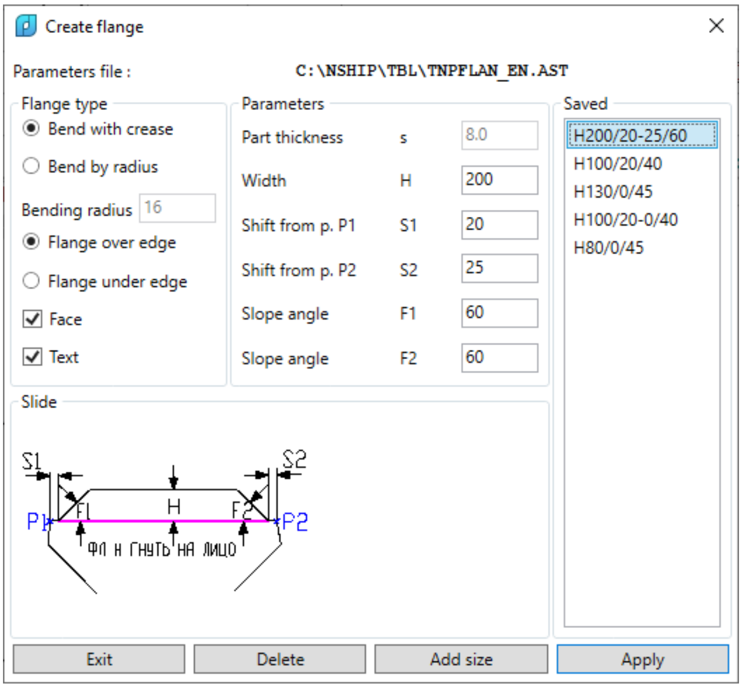
Command **Apply** inserts selected text into the drawing. Next in the cycle there is requested insertion of other inscription from the window **Bending Text**. Esc is used for exit from the cycle.

Buttons **Delete** and **Add** serve for changing contents of the file *NSHIP\Tbl\Tgibka_en.<shd>* with prepared texts.

Command **Max curvature arrow** (Fmax) creates service text on layer TIPDET in the form **Fmax=X**, where X is a value of maximal bending arrow for the part (this command is added for some customers).

Command **Templates quantity** (Nsh=) is similar to the previous one, but creates on TIPDET text **Nsh=X**, where X is a number of bending templates for manufacturing this part (customer wish).

Command **Flange** ( ) is used for changing part straight outer contour by adding flange geometry, e.g. for knees, brackets. It calls dialog box **Create flange** (pic. 110).




Pic. 110. Dialog box **Create flange**

Area **Flange type** define bending case by radio buttons (**Bend with crease** or **Bend by radius**), with value **Bend radius** in mm (default value is given by program, equal to 2*s if s<=8 mm and 3*s if s>8 mm, where s is part thickness). Radio buttons **Flange over edge** and **Flange under edge** serve for pointing out location of thickness relative to base part edge. Checkbox **Face** influences on bending direction: face side (to the observer) or back side (from the observer). Checkbox **Text** defines presence of bending text on layer TIPDET, with flange description.

Parameters are illustrated in the image area **Flange slide**.

Area **Parameters** contains values of parameters defining flange form:

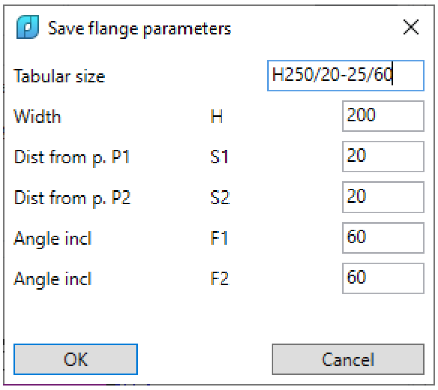
- **Part thickness s**, unfilled for a new part and filled if part was loaded by command **PART > Open part**;
- **Width H**, mm, flange width from base edge;
- **Shift from p. P1  S1**, distance inside part from the first point (P1) on edge;
- **Shift from p. P2  S2**, distance inside part from the second point (P2) on edge;
- **Slope angle F1**, angle, degrees, of the first flange side line inclination relative base edge;
- **Slope angle F2**, angle, degrees, of the second flange side line inclination relative base edge.

After filling in data in areas **Flange type** and **Parameters** clicking on button **Apply** closes window and launches procedure of flange creation and change of the part outer contour.

Parameters of repeated flange sizes can be saved in the system file *NSHIP\Tbl\Tnpflan_en.<shd>* (where *<shd>* is a 3-4 symbols designation of shipyard documentation code, e.g. *TST*, *BAL*).

Earlier saved flange sizes are displayed in the listbox **Saved**. If you select element in the list then its parameter values will be copied to area **Parameters**.

For saving size and its parameters use button **Add size**, click on which opens dialog box **Save flange parameters** (pic. 111).




Pic. 111. Dialog for saving flange size with new name

In this window user must fill in fields **Tabular size**, **Width H**, **Dist from p. P1**, **Dist from p. P2**, **Angle incl F1**, **Angle incl F2**. To save size click **OK**. Name entered in **Tabular size** must not coincide with previously saved size names. If name repetition is found then a message in the bottom side of the window will be shown, with red square at the beginning of the line. Inside names spaces and semicolons (;) are not allowed and will be removed automatically.

Button **Delete** (see pic. 110) is intended for removing earlier saved flange size. Button **Exit** closes window without flange building.

Dialog of command **Flange**:

*OUTER CONTOUR IS OK!*

*Pick first point on contour:*

*Pick second point on contour:*

*Side to build flange:*

*Part thickness:*

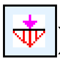
*Bending radius:*

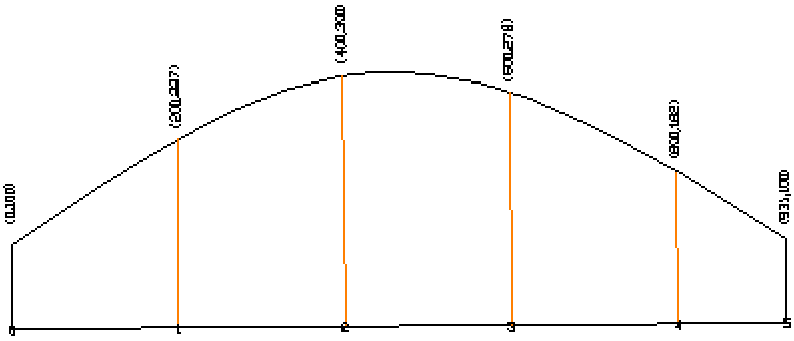
*OUTER CONTOUR IS OK!*

*Insertion point:*

*Rotation angle:*

Thickness and bending radius are requested if they are not filled in the window **Create flange**.

Command **Template** () of submenu **PART > BEND** is used for building bending template (template is an auxiliary tool for bending operation in the workshop). As a preliminary step it is necessary to build curvilinear POLYLINE that will be template boundary and to copy it to a free drawing place. Sample of created template is shown on dr. 112.



Pic. 112. Sample template tool

Dialog of command **Template**:

*Select edge to build a template:*

*Start point (p1) on line <exit>:*

*End point (p2) on line <exit>:*

*Enter template dulling value: positive or 0 - template base will be placed to the right of vector p1-p2, negative - to the left.*

*Enter number <100>:*

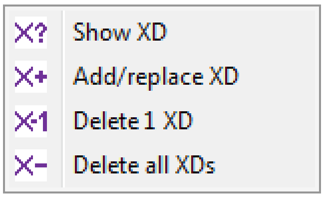
*Base length = 1121*

*Define step? [Y/N] <Y>:*

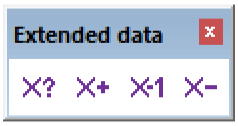
*Section step <200>:*

## 15 EXTENDED DATA

Some commands write extended data (xdata, XD) to the entities. These data can be later used for extracting specific information for part manufacturing (bend, create chamfer etc.). Xdata are invisible, but can be displayed, added or edited with commands of submenu **PART > Object XD** (pic. 113) and toolbar **Extended data** (pic. 114).

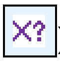



Pic. 113. Submenu **Object XD**

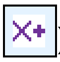



Pic. 114. Toolbar **Extended data**

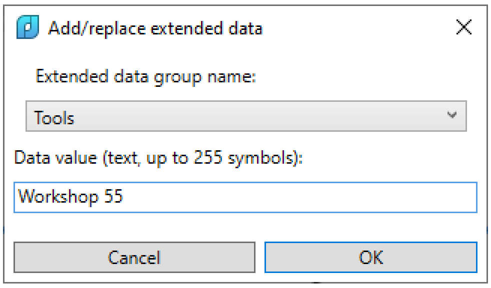
Extended data in entities are divided into groups which names are conditionally called application IDs and they are stored in the symbol table APPID inside DWG file. System **N-Ship** uses the following application names for its xdata: ALLOWANCE, GIBKA, R_EdgeHandling, Tehn_nadp, etc. List of groups can be increased and saved in the textual file *NSHIP\Ini\rd_groups.ini* and can be modified by user manually.

Command **Show XD** () requests objects and outputs entity DXF data including extended data, for example:

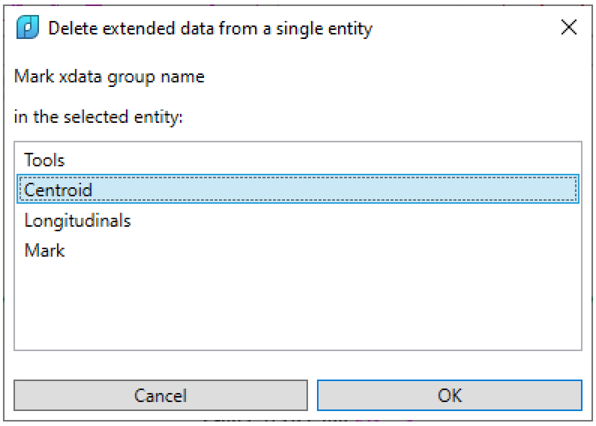
*Object 0: ((-1 . <Entity name: 7ffff705b40>) (0 . "MTEXT") (330 . <Entity name: 7ffff7039f0>) (5 . "234") (100 . "AcDbEntity") (67 . 0) (410 . "Model") (8 . "FASKA") (62 . 7) (100 . "AcDbMText") (10 172.659 75.339 0.0) (40 . 5.2) (41 . 0.0) (46 . 0.0) (71 . 7) (72 . 5) (1 . "\\A1;Л30%%d пр4") (7 . "Standard") (210 0.0 0.0 1.0) (11 0.855556 0.517711 0.0) (42 . 22.0273) (43 . 6.66494) (50 . 0.544173) (73 . 1) (44 . 1.0) (-3 ("R_EdgeHandling" (1000 . "{240.000,155.000}{240.000,44.179}{240.000,0.000} Name FaceChamfer Corner 30.0 Dulling 4.0 LengthEdge 155.000 Concavity 1"))))*

Command **Add/replace XD** () requests entities and suggests to define application name. User must enter a string that will be saved in entity's extended data with DXF codes 1000 and greater and will be bound with application of the defined name. Dialog box **Add/replace extended data** (pic. 115) is opened.

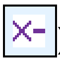
Command **Delete 1 XD** () requests a single entity and name of application which xdata should be removed from the selected entity. Dialog box **Delete extended data from a single entity** (pic. 116) is used for this operation.




Pic. 115. Window **Add/replace extended data**




Pic. 116. Window **Delete extended data from a single entity**

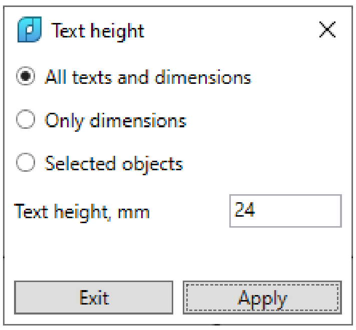
Command **Delete all XDs** () requests entities and deletes from them **all** the extended data.

## 16 CHANGING TEXT HEIGHT

Command **Text height** is called by these tools:

- menu item **PART > Text height**,
- button  of toolbar **Technology**.

Command opens dialog box shown on pic. 117.




Pic. 117. Window **Text height**

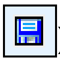
Radio buttons are used for selection of change mode: **All texts and dimensions**, **Only dimensions** or **Selected object**. Field **Text height, mm** shows default value. User can edit this value and click button **Apply**.

Button **Exit** closes window with no action.

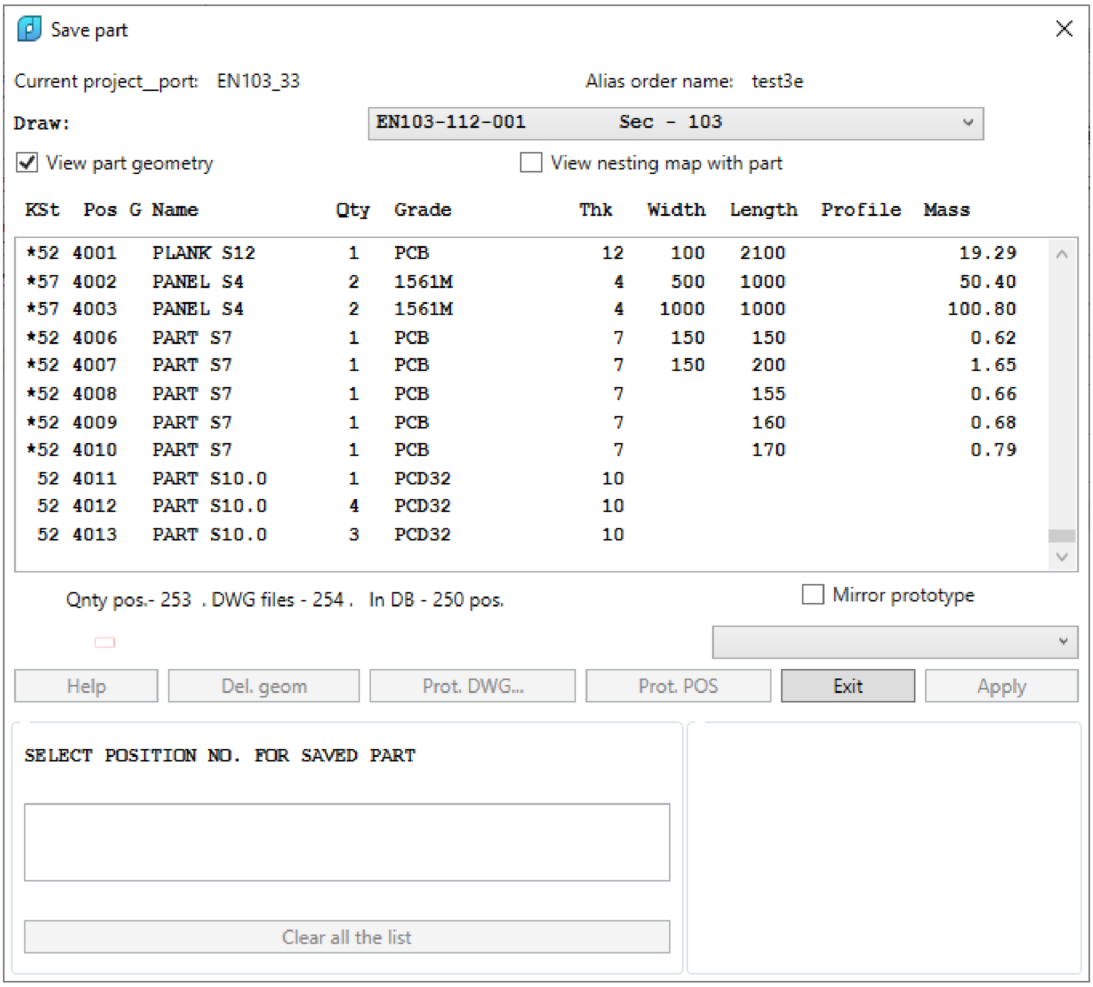
## 17 SAVING PART AND CREATION OF TNC, FPD

System generates and saves part sketches as DWG files and special part forms for workshops (called TNCs or FPDs).

### Saving parts

Command **PART > Save part** () saves part calculated parameters (area, weight, etc.) to DB table *specp.dbf* to folder *Dbf* and writes part geometry into DWG file to folder *Dwg*, both inside folder of current project_port.

If part was created without command **PART > Open part**, then its position number is still undefined and command **Save part** opens dialog box **Save part** with prompt **SELECT POSITION NO. FOR SAVED PART** (pic. 118).



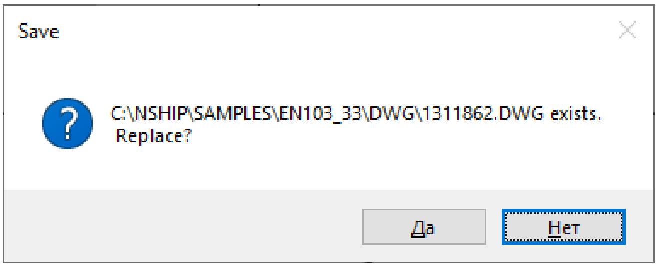

Pic. 118. Dialog box for selection of position number

In the upper zone of the window there are shown current project_portion, vessel (order) name and draw. Drop-down list **Draw** contains list of draws for changing draw is necessary. Adter changing the draw, list of part positions (specification) is reloaded.

Positions already having geometry DWG file in the folder *Dwg* of the current project_port are marked with asterisk (*).

After selection of position in the central list box button **Apply** becomes enabled. On clicking this button program closes window and saves part geometry sketch from screen and part attributes to DB table *specp.dbf*.

If position already has DWG file then there will be request for replacement, as in pic. 119.




Pic. 119. Request for DWG replacement

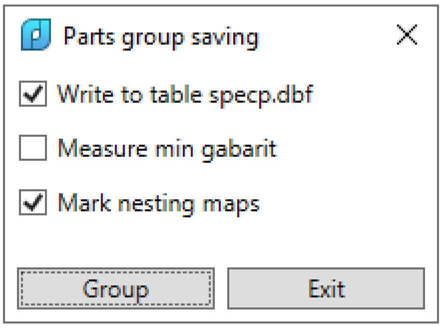
Dialog of command **Save part**:

*Pos. 1 Sec. 104 Draw EN103-112-104 Saving...*

After saving current drawing is closed but system is ready for next operations (for example, **PART > New part** or **PART > Open part**).

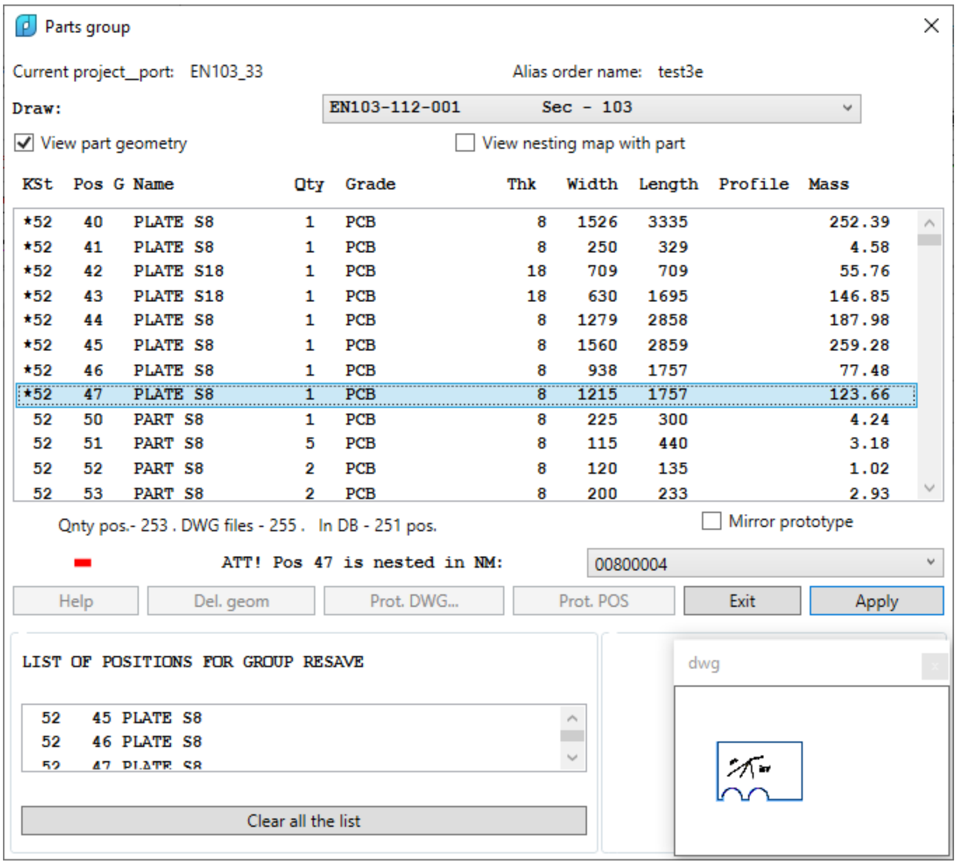
## Saving group of parts

Command **PART > Save group** is used for rewriting of parts group (e.g. requiring change of part label form). Command opens dialog box on pic. 120.




Pic. 120. Dialog box for resaving group of parts

Click on button **Group** opens dialog box for selecting positions (pic. 121).



Pic. 121. Dialog **Parts group**

User must select parts for the group and click **Apply** button. Program will open DWG files of selected positions and rewrite one by one.

Saving part documents in forms

Before transferring parts drawings to the workshop usually special A4 part forms are filled in additional specific drawings. They are called either Technologic norming cards (TNC), or Marshroute technologic cards (MTC), or Forms for print documents (FPD). We use ‘TNC’.

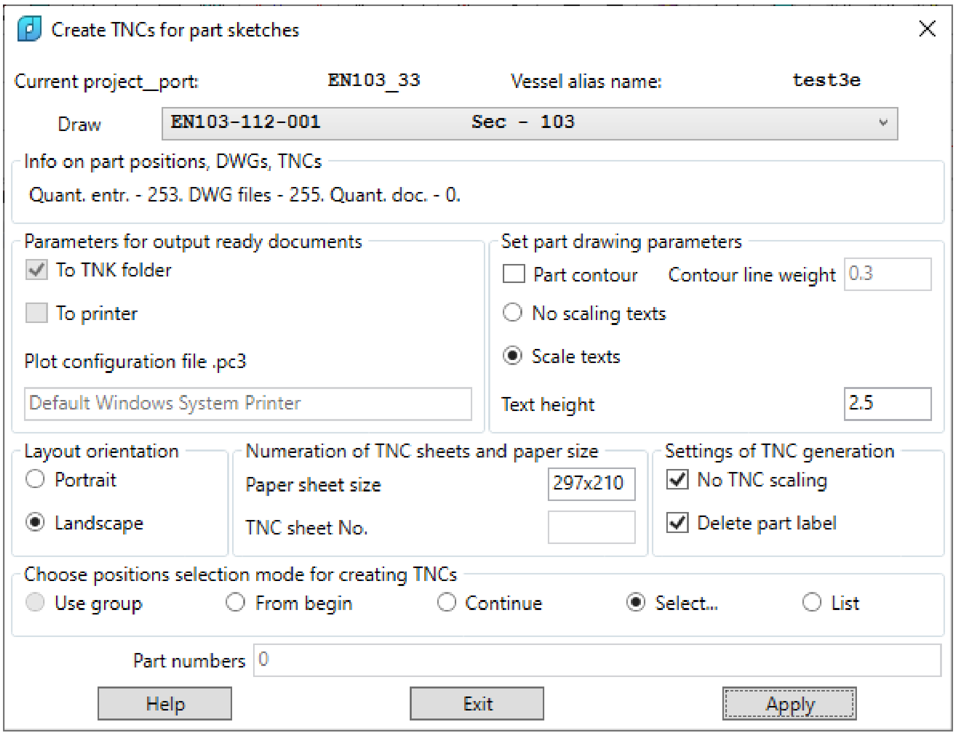
Commands **PART > Create TNC**, **Create TNC_2** serve for creating TNCs for workshop. The second command works faster as some features are cut.

Command **Create TNC** opens dialog box shown on pic. 122.

This window serves for setting parameters of creating part sketch TNC documents. Checkbox **To TNK folder of order** in area **Parameters for output ready documents** is always set on and disabled.

If user intends to print TNCs then the most comfortable way is to run TNCs calculation and to print them with command **BDATA > Print DWGs from folder > to system printer**.

Area **Set part drawing parameters** can define additional thickening for part outer contour line with setting checkbox **Part contour** and gives thickness (lineweight) value in **Contour line weight**. Contour lineweight is displayed when mode button **LWT** is pressed in the status line of nanoCAD.



Pic. 122. Dialog box for creation of TNCs

Radio button **Scale texts** allows to apply the same height to all texts. Value must be entered in the box **Text height** (is not applied to multitexts of chamfers and bevels).

Area **Settings of TNC generation** includes checkbox **No TNC scaling**. It is used when part contour is not scaled while inserting into A4 table form (otherwise reverse scaling will be applied only to table form). Setting checkbox **Delete part label** means that part label will be removed from the document. If checkbox is cleared then label will stay in TNC.

Area **Choose positions selection mode for creating TNCs** contains radio button **From begin** that corresponds to deleting all old TNCs of the current draw and recalculating all TNCs by pressing button **Apply**.

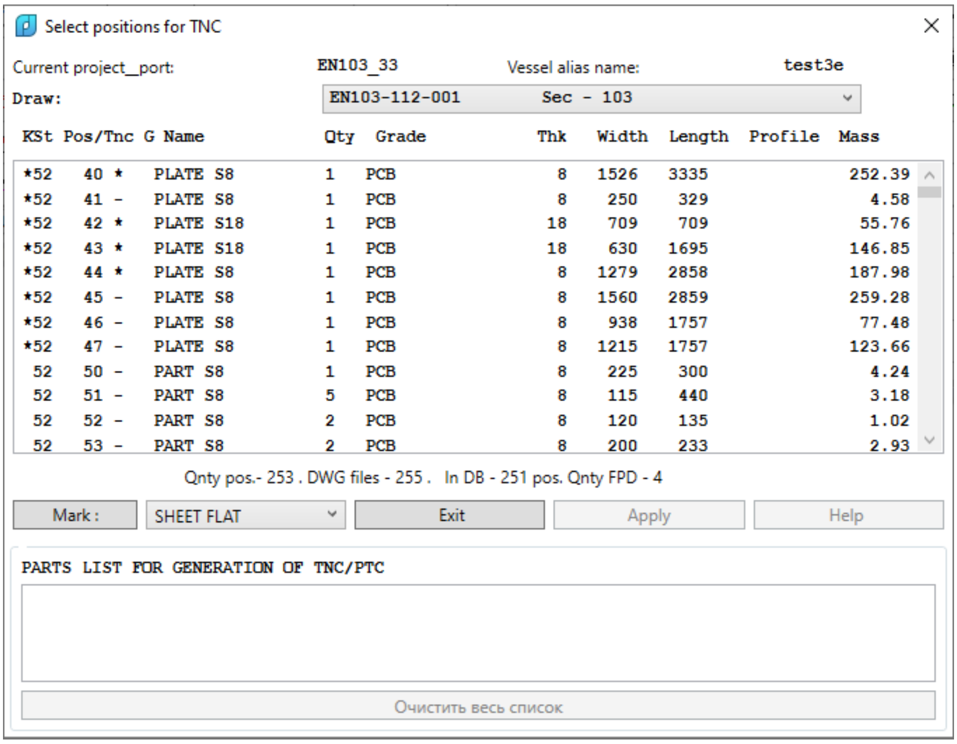
Radio button **Continue** is used for creating missing documents for the draw.

Radio button **Select** with button opens dialog box with parts list where user must select those positions for which program must recalculated TNCs (pic. 123).

In the list of positions for the draw selected in the **Draw** drop-down list there is an asterisk sign (*) in the **Tnc** column – it means that for this position TNC DWG file is already generated.

Click on position line forms lower area **PARTS LIST FOR GENERATION OF TNC/PTC**. There are also filters list (**SHEET FLAT**, **All**) and button **Mark:** to select positions by material type. Double click on the line in the lower area deletes clicked position.

After completing the lower list user must click **Apply** button.



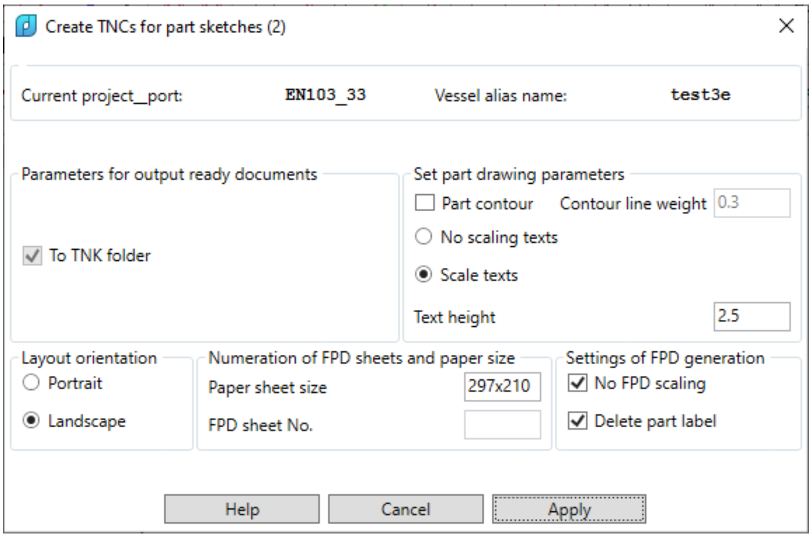

Pic. 123. Dialog box **Select positions for TNC**

In the dialog box **Create TNCs for part sketches** if radio button **List** is activated then box **Part numbers** is used for defining list of positions for which TNCs must be created. Use comma as divider for number intervals. For example: 20,32-35,43-52.

By click button **Apply** user launches procedure of creating TNCs. Button **Exit** closes dialog box with no actions.

All the DWG files created in the *Tnk* folder can be converted to PDF by command **BDATA > Print DWGs from folder > to PDF**. To print all the files from the *Tnk* folder use nanoCAD batch print command.

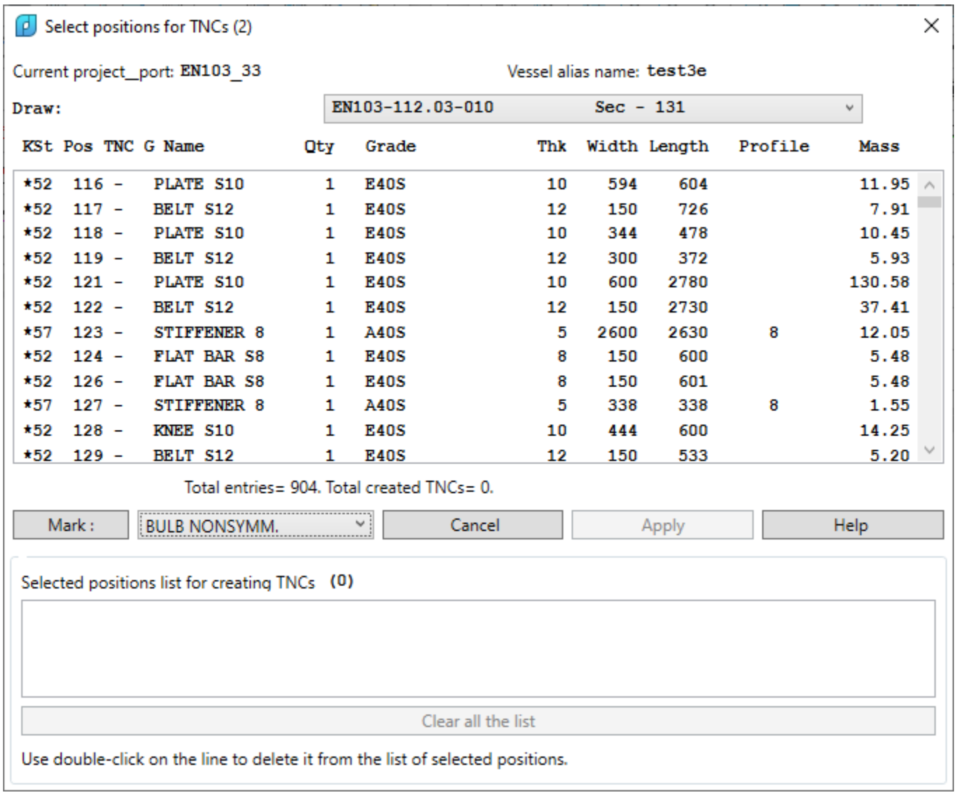
Command **PART > Create TNC_2** opens window **Create TNCs for part sketches (2)** (pic. 124).




Pic. 124. Dialog box **Create TNCs for part sketches (2)**

In comparison with the window **Create TNCs for part sketches** there is no area **Choose positions selection mode for creating TNCs** and the only method is selection of positions in the parts list.

Click on button **Apply** goes to dialog box **Select positions for TNC (2)** (pic. 125).




Pic. 125. Dialog box **Select positions for TNC (2)**

Work with this dialog is similar to work with the dialog **Select positions for TNC**. Positions can be selected by individual click on its line or by filter (all material types used in the draw are shown, e.g. **BULB NONSYMM.**, **PLATE FLAT** etc.) and button **Mark:**.

### Part technological parameters

If checkbox **Create data for norming and workshop handling** is set in attributes settings dialog then additional geometry & technology parameters are calculated to be applied for developing part handling route in the workshop.

**Note.** It is also necessary to exist non-empty table *sign_par.dbf* in the folder *NSHIP\Plants_settings\<shipyard>\Autotehno*. Table contains parameter types list.

Additional parameter values are calculated while saving part DWG file and are stored in the table *sign_par_obj.dbf* that is included into project_port DB.

Parameters are divided into groups. After calculation corresponding messages are output in the command line, e.g.:

*Parameters0: SS=7 LL=0.23 BB=0.23 EFA=yes ELA=no EGI=no ETO=yes EHO=no TMA=sheet PVV=0.0 MRK=PCB CNT=straight COC=non convex ESK=no WPA=1.688 PNK=0.93 KOL=1*

*Parameters1: ETO=yes EPG=yes EGI=yes TPR=assembly TPR=assembly TPR=bend TPR=bend TPR=bend TPR=bend TPR=bend EFA=yes TFA=face AFA=30 BFA=2 LFA=0.09 FOF=concave ELA=no ESD=no ESV=no EHO=no ESK=no*

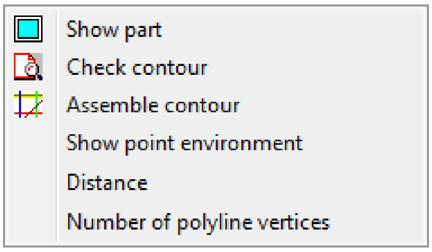
*Parameters2: LRA=0 KKP=2 KLG=0 KDL=0 RLK=0 DLK=0 KLN=0 PRG=0 NSH=0*

By form parameters can be checkboxes (yes, no), types (sheet, bending etc.) and values (length, quantity etc.).

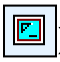
Parameters can be used by the procedures of setting route of part handling in the workshop.

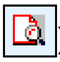
## 18 SERVICE

Submenu **PART > Service** is shown on dr. 126.




Pic. 126. Submenu **Service**

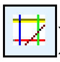
Command **Show part** () moves part to screen center.

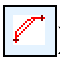
Command **Check contour** () starts verification of part outer contour (it must be closed 2D-POLYLINE and unique on layer KBAS). If check failed then contour changes its color to red and program issues warning. In this case user must run command **Assemble contour**. Command requests:

*Specify point inside contour:*

If no problems are found the there is a message:

*BASE CONTOUR IS GOOD !*

Command **Assemble contour** () launches some operations for uniting outer contour and contour hole lines. If assembling fails then contour remains unchanged and warning appears. In that case use must do manual assembling with graphical editor command PEDIT.

Command **Simplify contour** () allows as possible to reduce number of contour polyline points and preserve geometry inside given accuracy. The command is unrecommended for multiple running because summary changes can strongly change the part contour (close to straight).

Command **Show point environment** runs zooming and sets window near to required point.

Command **Distance** allows measuring distance by straight line between two points.

Command **Number of polyline vertices** is used for displaying number of vertices:

*Select polyline:*

*Quantity of vertices for this polyline: 14.*

Note. The command works only with 2D-POLYLINE objects. If object is of type LWPOLYLINE it is necessary to convert to 2D-POLYLINE (with command of submenu **PART > CONTOUR > POLYLINES**).