

Progressive Dies Extension

Customization of Components

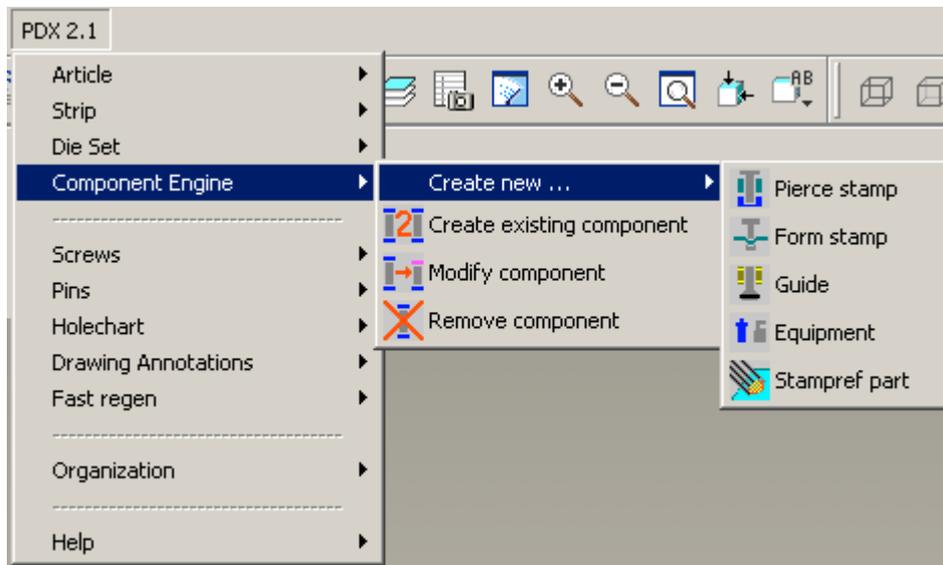
1 Structure of Component Library

The component library of Progressive Dies Extension is stored in directory $\langle PDX \rangle \setminus \langle UNIT \rangle \setminus component_engine$, where $\langle PDX \rangle$ is the installation folder of Progressive Dies Extension and $\langle UNIT \rangle$ is either “mm” for metric or “inch” for imperial units.

The components are organized in four different groups which reflect their functionality:

- Pierce stamps in sub directory *cut_stamps*
- Form stamps in sub directory *form_stamps*
- Guides in sub directory *guides*
- Equipment in sub directory *accessories*

In the figure below the corresponding menus are shown. Only the stamp ref part is not part of the component engine.



Each component and its logic is stored in one of these four folders or in a sub folder of them. The following files will be found in these directories:

- Pro/ENGINEER files (parts, assemblies, UDFs, etc): These are the templates of the library parts
- Graphic files (*.gif): Graphics used for display
- sel_list.txt: Describes the structure of the library
- component description files (*.tab) Contains assembling and configuration logic and GUI
- param_relations.txt Describes parameters for bill of material

1.1 Graphic files

Graphic files are regular gif files. They could be created with different third party software packages. Graphic files shown when choosing the component are scaled to 100 pixel in height. For avoiding quality loss in re-scaling use images with height 100 pixels.

1.2 sel_list.txt

The file *sel_list.txt* defines the type of the component. Parts, assemblies, UDFs or sub directories could be described.

Each line consists of *name* and *option*. The following options are available:

- dir
- part
- part_copy
- asm
- asm_copy
- udf

The option *dir* declares a new sub folder with the specified name.

To assemble a Pro/ENGINEER part the option *part* is supplied. In this case the part will be assembled by its regular name. If a copy of this part with a new name created by a consecutive number is to be assembled, the option *part_copy* should be used.

asm and *asm_copy* offer the same functionality for Pro/ENGINEER assemblies. Finally the option *udf* declares an Pro/ENGINEER user defined feature.

Typically the options *part* and *assembly* are used for describing standard supplier components that should be assembled.

If a graphic file with the same name is available, it will be displayed in the component selection dialog.

The lines below demonstrate some example entries in the *sel_list.txt* file:

```
sample dir          Defines a subdirectory named sample
screw part         Defines a component screw.
example asm_copy   Defines an assembly which is to be copied
```

Exercise:

- Create a new subdirectory for cut stamps with the name *sample*

1.3 param_relations.txt

This file contains bill of material information. The string parameter *PDX_PART_NAME* and *PDX_ORD_NUMBER* parameter are set up as relations. Typically they will be built by the parameters *BUW_NAME*, *BUW_TYPE* and *BUW_SIZE* which are usually provided at a component.

This file is available for all main branches of *component_library* (*cut_stamps*, *form_stamps*, *guides*, *accessories*). It could also be specialized directly in the directory which contains the component.

1.4 Component description files (extension *.tab)

Component description files could contain four different types of information – assembly description, graphical user interface, table description and component description. Each block is introduced and completed by specific tags which are shown below:

- Assembly description block: `BEGIN_ASM_DESCR ... END_ASM_DESCR`
- Graphical user interface block: `BEGIN_GUI_DESCR ... END_GUI_DESCR`
- Table description block `BEGIN_TAB_DESCR ... END_TAB_DESCR`
- Component description block: `BEGIN_COMP_DESCR ... END_COMP_DESCR`

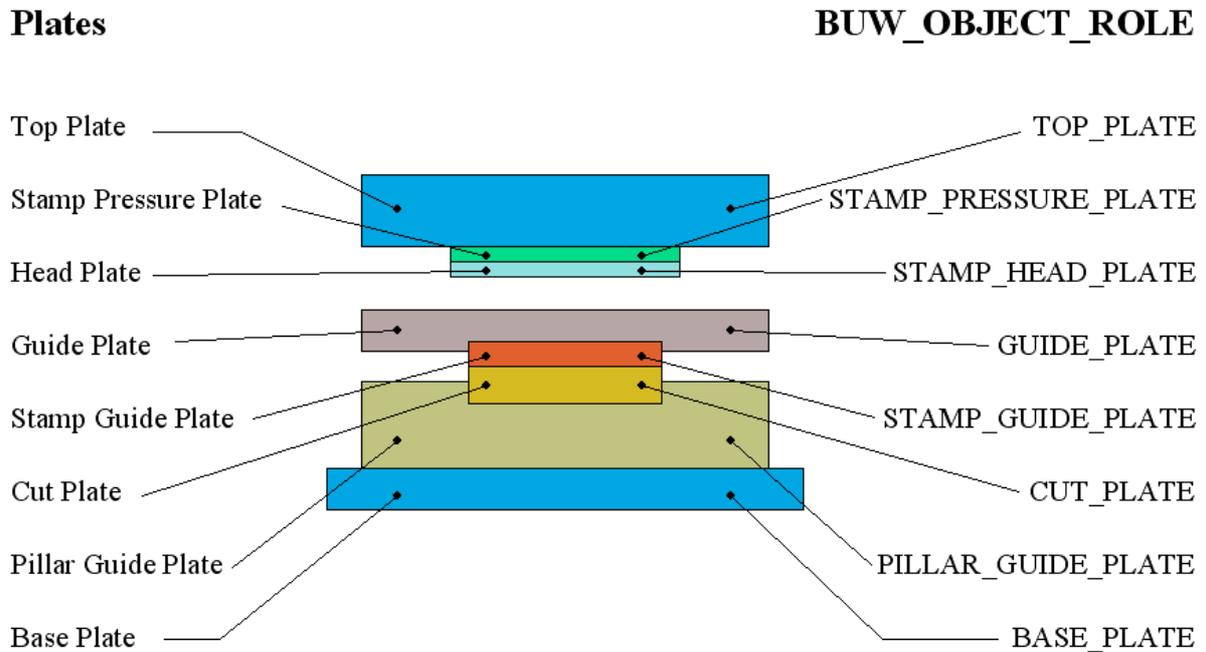
Beginning with chapter 3 the meaning for each block is explained in detail. All examples are provided in *tutorial_examples.zip*.

Important:

This document does not replace the administration guide. It shows how to handle different problems by the component engine, but does not provide all details for each command. Please use administration guide for further reference.

2 Tooling structure

The following image shows the tooling structure which is used for the example. On the left side the plate names are listed and on the right side the object role. This object role is stored in the string parameter *BUW_OBJECT_ROLE* for each plate. This parameter provides a suitable way to distinguish between the different plates.



Furthermore the top and bottom surface of each plate has the name *TOP* and *BOTTOM*. These names could later on be used to search a specific side of the plate. *WIDTH*, *LENGTH* and *THICKNESS* parameters are supplied to each plate as dimension. Some components in the library figure out the thickness by using this dimension.

3 Creating a simple cut stamp

In this chapter a simple rectangular cut stamp *cut_stamp.prt* will be created from the scratch. Configuration, assembling, cut outs, drawings, bill of material information will be shown in different steps.

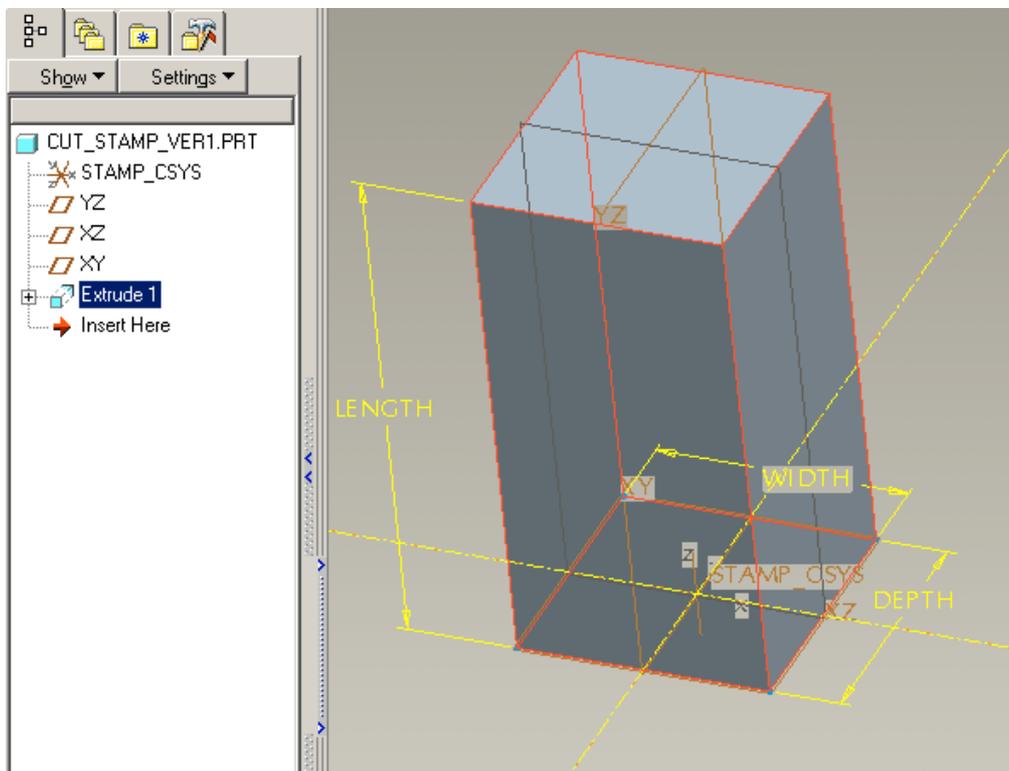
For testing the component, please create the standard die set “demo_buw_progdie” and the demo strip.

3.1 Create Pro/ENGINEER part

In the first step a Pro/ENGINEER part for the cut stamp will be created and saved to a new subfolder inside the component engine.

Exercise:

- Create Pro/ENGINEER part called *cut_stamp.prt* in folder <PDX>\mm\component_engine\cut_stamps\sample
- Create standard coordinate system *STAMP_CSYS* and standard datum planes *YZ*, *XZ*, *XY*.
- Create solid extrude sketched on *XY* plane in *z* direction.
- Apply dimensions with symbolic name *LENGTH* (=80.0), *Width*(=20.0), *DEPTH* (=20.0).



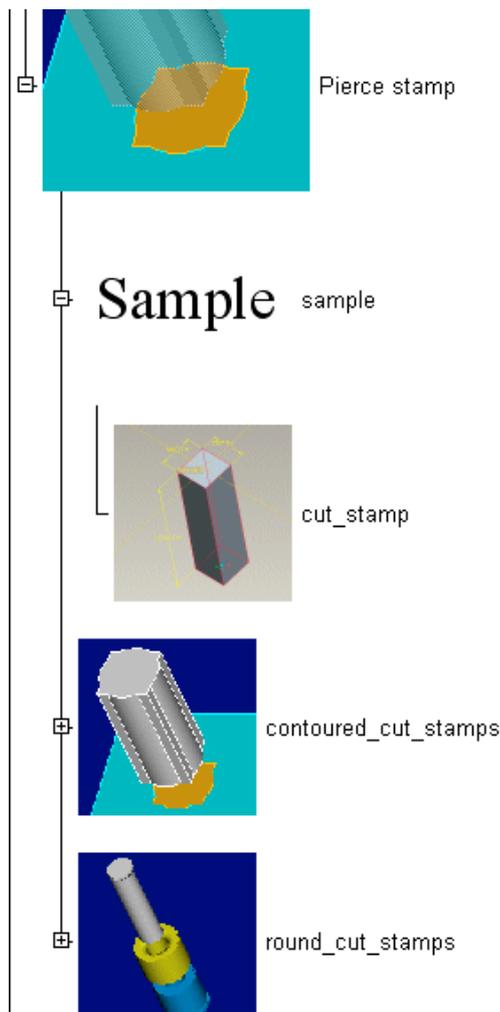
3.2 Integrate new component in library

Exercise:

- Add the following line to file <PDX>\mm\component_engine\cut_stamps\sel_list.txt
sample dir
- Create a gif image called *sample.gif* and copy it to directory <PDX>\mm\component_engine\cut_stamps. This gif image will be shown for the folder *Sample*. The image will be re-scaled to height of 100. To avoid quality loss try to create the gif with height 100.
- Create a gif image called *cut_stamp.gif* and copy it to directory <PDX>\mm\component_engine\cut_stamps\sample. This gif image will be shown when selecting the component. The image will be re-scaled to height of 100. To avoid quality loss try to create the gif with height 100.

Result:

- Select *PDX21 -> Component Engine -> create new ... -> Pierce stamp*. You should see the following dialog



3.3 Provide graphical user interface for cut stamp

In this step the first component description file will be created. At first it will mainly consists of the user interface definition. Aspects of assembling are appended in later stages.

Exercise:

- Create a gif image called *cut_stamp_tab.gif* and copy it to directory <PDX>\mm\component_engine\cut_stamps\sample. This gif image will be shown in the graphical user dialog for the component. The image will not be scaled.

- Create a sel_list.txt for this directory with the following line as content:

```
cut_stamp part_copy
```

- Create new file *cut_stamp.tab* with text editor in directory <PDX>\mm\component_engine\cut_stamps\sample

- Write the following lines to this file

```
BEGIN_GUI_DESCR

    GLOBAL_PICTURE cut_stamp_tab.gif

    USER_SELECT CSYS ASM_CSYS 1

    USER_INPUT_PARAM DOUBLE STAMP_WIDTH
    USER_INPUT_PARAM DOUBLE STAMP_DEPTH
    USER_INPUT_PARAM DOUBLE STAMP_LENGTH

END_GUI_DESCR

BEGIN_ASM_DESCR

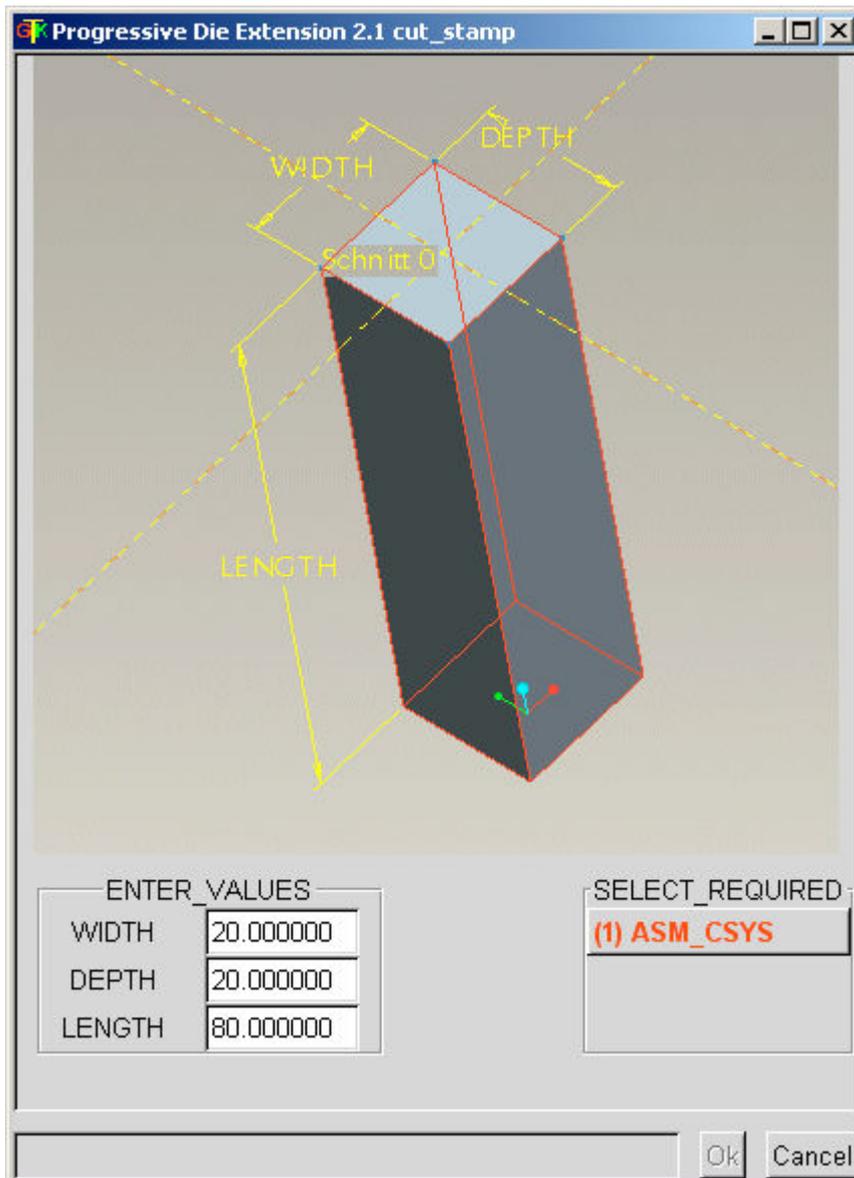
    SEARCH_MDL_PARAM THIS WIDTH STAMP_WIDTH
    SEARCH_MDL_PARAM THIS DEPTH STAMP_DEPTH
    SEARCH_MDL_PARAM THIS LENGTH STAMP_LENGTH

    CONFIG_ELEM

END_ASM_DESCR
```

Result:

- After processing the tab file the following user dialog appears. The standard values for *WIDTH*, *DEPTH* and *LENGTH* correlates to the dimensions you have created in the *cut_stamp.prt*.
- After pressing the *OK* button the component will just be retrieved in session will not be assembled.



Information:

- Processing of the component description files starts in the assembling information block after the tag *BEGIN_ASM_DESCR*.
- The three lines starting with *SEARCH_MDL_PARAM* command search the stamps for the different dimensions and store the result to a variable in the component program.
- The next line *CONFIG_ELEM* initiates the system to process the graphical user interface.
- Graphical user interface is beginning with *BEGIN_GUI_DESCR* and ending on *END_GUI_DESCR*.
- First line *GLOBAL_PICTURE* in GUI defines the gif image which will be shown in the dialog mask.
- *USER_SELECT* line defines the button for selecting a coordinate system in the dialog.

- *USER_INPUT_PARAM* defines the text fields for the dimension values. Since the variables (*WIDTH*, *DEPTH*, *LENGTH*) are already filled with values via *SEARCH_MDL_PARAM* the dimension values for the *cut_stamp.prt* are shown.
- After pressing the OK button in the dialog the mask is closed and modified values are updated in the *cut_stamp.prt*. The *BEGIN_GUI_DESCR ... END_GUI_DESCR* block is completely processed and the next line of *BEGIN_ASM_DESCR ... END_ASM_DESCR* is to be executed.
- Since no line in *BEGIN_ASM_DESCR ... END_ASM_DESCR* follows the *CONFIG_ELEM* statement, the component procession has finished.
- For more information on each command please see the Progressive Dies Admin Guide.

Exercise:

- modify the *WIDTH*, *DEPTH* and *LENGTH* values in the dialog mask and watch the changes in the *cut_stamp.prt*.

3.4 Assemble the part

In the previous chapter the *cut_stamp.prt* was configured, but it was not defined how this component should be assembled to the Die Set. Now a selected coordinate system should be used for assembling.

Exercise:

- Add the following lines to the end of the assembly description block

```
SEARCH_MDL_REF THIS CSYS STAMP_CSYS STAMP_CSYS

ASSEMBLE THIS
    CSYS ASM_CSYS STAMP_CSYS
END_ASSEMBLE
```

Information:

- The *SEARCH_MDL_REF* command searches the component for a coordinates system *STAMP_CSYS*. This coordinate system is used for assembling in the next lines.
- *ASSEMBLE THIS ... END_ASSEMBLE* contains the constraints. In this case only one constraint is required.

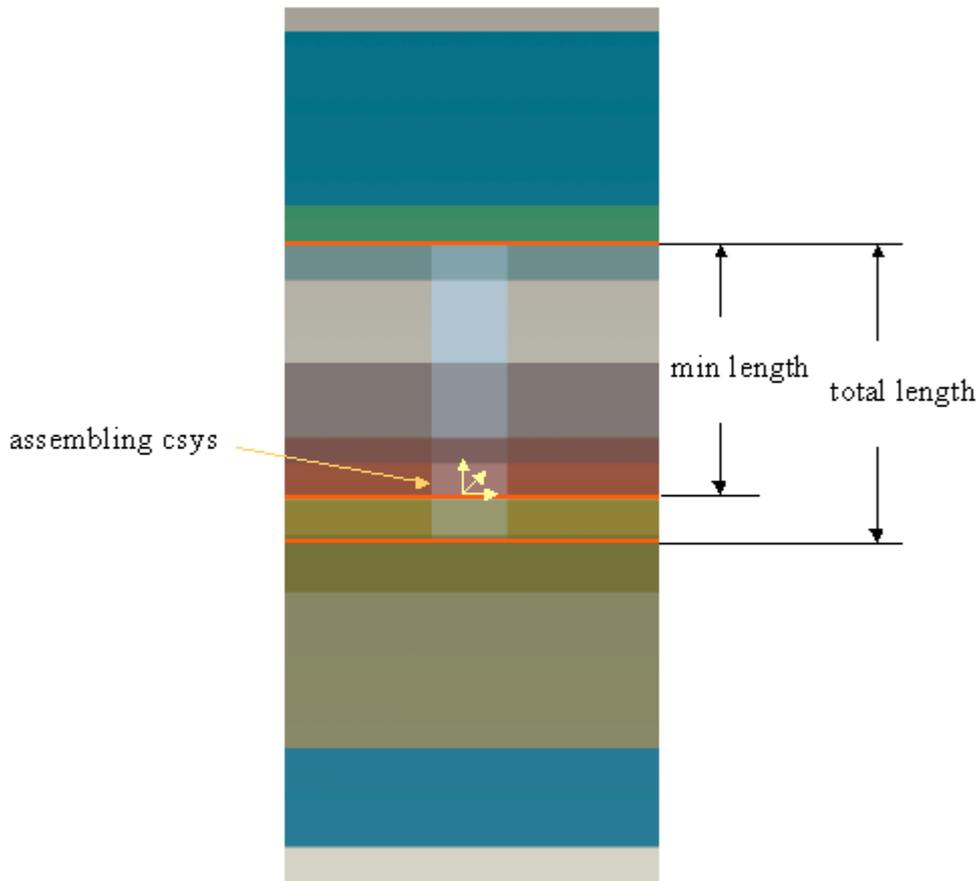
Each time you assemble a new cut stamp a new version will be created. The project shortcut prefix will be added to the name and a consecutive number is appended at the end.

Exercise:

- Modify the *part_copy* to *part* in the *sel_list.txt* and test to assemble the component several times. What happens when dimensions are changed?

3.5 Improve assembling

In the next step the assembling should become more comfortable. Even though the stamp is assembled to a coordinate system on strip, it should be assured that the stamp is aligned to the stamp pressure plate (see figure below). The minimal required length for the stamp should be measured as a hint for the user. Since it is very time consuming to create a coordinate system for each stamp, only a point should be used to determine the position. This point and other references in the Die Set will be used as references for a UDF coordinate system.



Exercise:

- Create a UDF called *STAMP_REF* of a coordinate system placed on a datum point and oriented to *A_XZ*, *A_YZ* datum planes in the tool. *Z* should be oriented in tool opening direction.
- Placement references for the UDF should be named *REF_POINT*, *REF_PLANE_XZ*, *REF_PLANE_YZ*.
- Save the UDF to folder <PDX>mm\component_engine\cut_stamps\sample.

Even when the plates thickness is changed, the stamp should remain aligned to stamp pressure plate. Therefore a UDF must be created which measures the distance between a datum point and a surface. This could be done by an analysis feature.

Exercise:

- Create a UDF called *MEASURE_DIST_POINT_FACE* which contains an analysis feature for measuring distance between datum point and surface.
- References should be called *REF_POINT* and *REF_FACE*.
- The analysis feature should create a parameter called *DISTANCE*.
- Save the UDF to folder <PDX>mm\component_engine\cut_stamps\sample.

Since measuring and point selection should be supplied to the user interface, the description has to be updated.

Exercise:

- Modify the user interface description in tab file. The new lines are marked bold. Remove the lines which are non longer required (selection of the coordinate system).

```
GLOBAL_PICTURE cut_stamp_tab.gif
```

```
USER_SELECT POINT POINT 1  
USER_SELECT FACE STAMP_PRESSURE_PLANE 2
```

```
USER_INPUT_PARAM DOUBLE STAMP_WIDTH  
USER_INPUT_PARAM DOUBLE STAMP_DEPTH  
USER_INPUT_PARAM DOUBLE STAMP_LENGTH
```

```
SHOW_PARAM DOUBLE MIN_LENGTH
```

```
SEARCH_STAMP_REF POINT BOTTOM STAMP_PRESSURE_PLATE STAMP_PRESSURE_PLANE
```

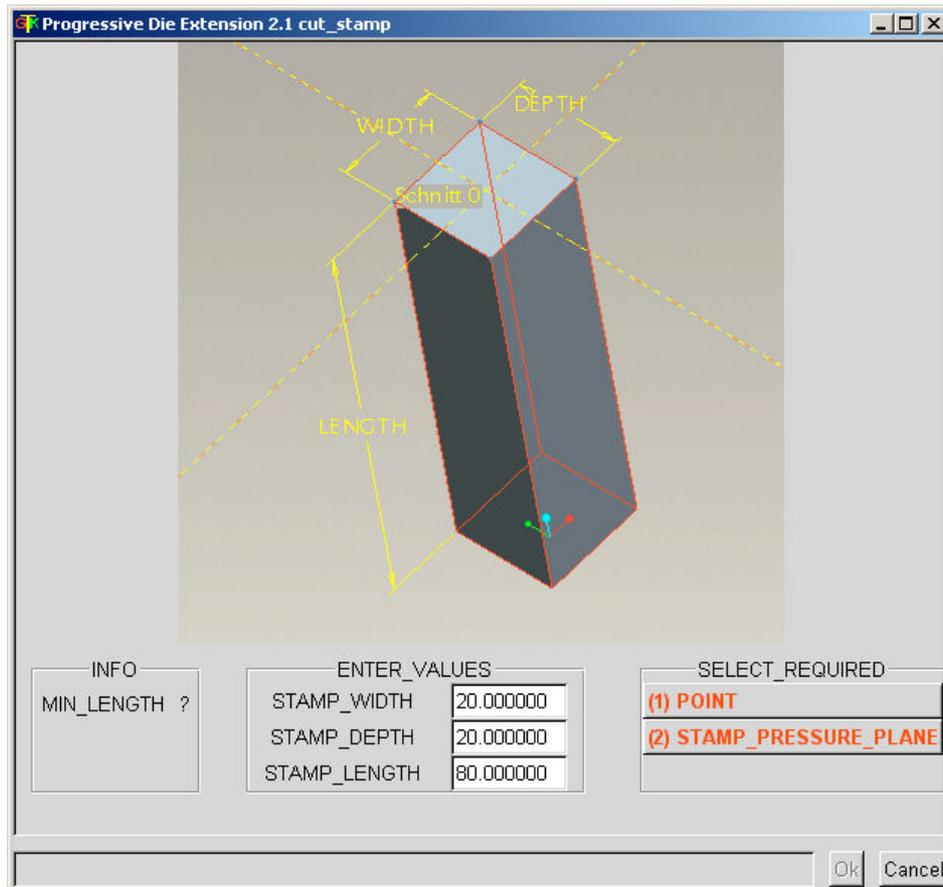
```
MEASURE_DISTANCE POINT STAMP_PRESSURE_PLANE MIN_LENGTH
```

Information:

- The first two new lines allow the user to select an datum point and a surface. These are required for assembling and for measurement.
- *SHOW_PARAM* line supplies an information parameter to the dialog.
- *SEARCH_STAMP_REF* is searching in Z direction for the bottom side (surface name in Pro/ENGINEER is *BOTTOM*) of the stamp pressure plate. The next line measures the distance between detected surface selected datum point.

Result:

- When executing the component, it will show up the following dialog:



As next step the assembling description has to be updated:

Exercise:

- Modify the assembling description according the following lines. New lines are marked in bold:

```

SEARCH_MDL_PARAM THIS WIDTH STAMP_WIDTH
SEARCH_MDL_PARAM THIS DEPTH STAMP_DEPTH
SEARCH_MDL_PARAM THIS LENGTH STAMP_LENGTH

SEARCH_MDL_REF THIS CSYS STAMP_CSYS STAMP_CSYS

CONFIG_ELEM

SEARCH_MDL_REF ASSEMBLY PLANE A_XZ A_XZ
SEARCH_MDL_REF ASSEMBLY PLANE A_YZ A_YZ

CREATE_UDF STAMP_REF ASSEMBLY
  UDF_REF REF_POINT POINT
  UDF_REF REF_PLANE_XZ A_XZ
  UDF_REF REF_PLANE_YZ A_YZ
  UDF_EXP_REF REF_CSYS CSYS 0
END_CREATE_UDF

CREATE_UDF MEASURE_DIST_POINT_FACE ASSEMBLY MEASURE_UDF
  UDF_REF REF_POINT POINT
  UDF_REF REF_FACE STAMP_PRESSURE_PLANE
END_CREATE_UDF

ASSEMBLE THIS_MEASURECOPY
  MEASURE_UDF DISTANCE
  CSYS REF_CSYS STAMP_CSYS
END_ASSEMBLE

```

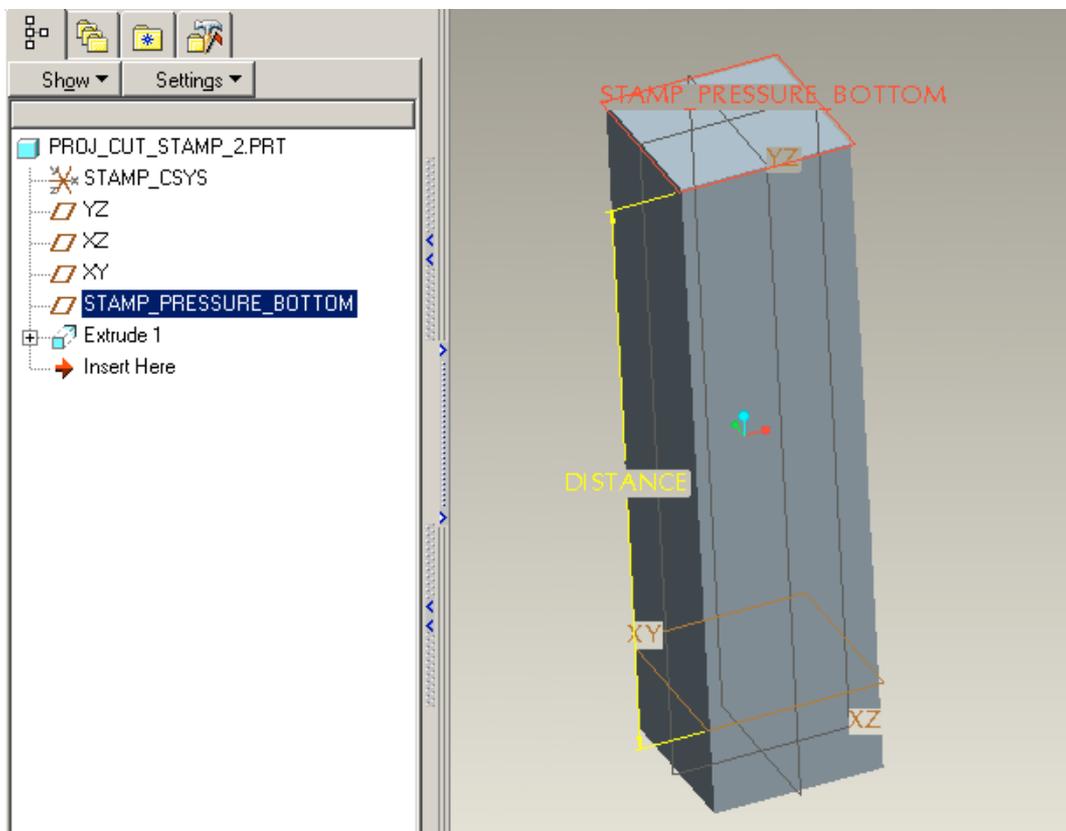
Information:

- The first new *SEARCH_MDL_REF* lines are searching for references in the project assemblies. Note Instead of *THIS* as search target *ASSEMBLY* is used. See Progressive Dies Admin Guide for more information.
- Subsequent the new UDFs are created as assembling reference and for measuring. The measurement result *DISTANCE* should be used by the *cut_stamp.prt*. Therefore the dimension *DISTANCE* needs to be established in Pro/ENGINEER model.
- *ASSEMBLE THIS_MEASURECOPY ... END_ASSEMBLY* envelops the assembly constraints. Option *THIS_MEASURECOPY* instead option *THIS* tells the system that the first line describes the parameters which should be added to the component via relation. In this case the value *DISTANCE* from UDF *MEASURE_UDF* is used.

Afterwards the component for the *cut_stamp* has to be modified:

Exercise:

- Add a new plane *STAMP_PRESSURE_BOTTOM* as offset to *XY* PLANE with distance 70. Rename the dimension name for offset to *DISTANCE*
- Redefine the extrusion feature. Locate sketch to *STAMP_PRESSURE_BOTTOM* and reverse the direction.



- Assemble the stamp, modify plate thickness and regenerate.

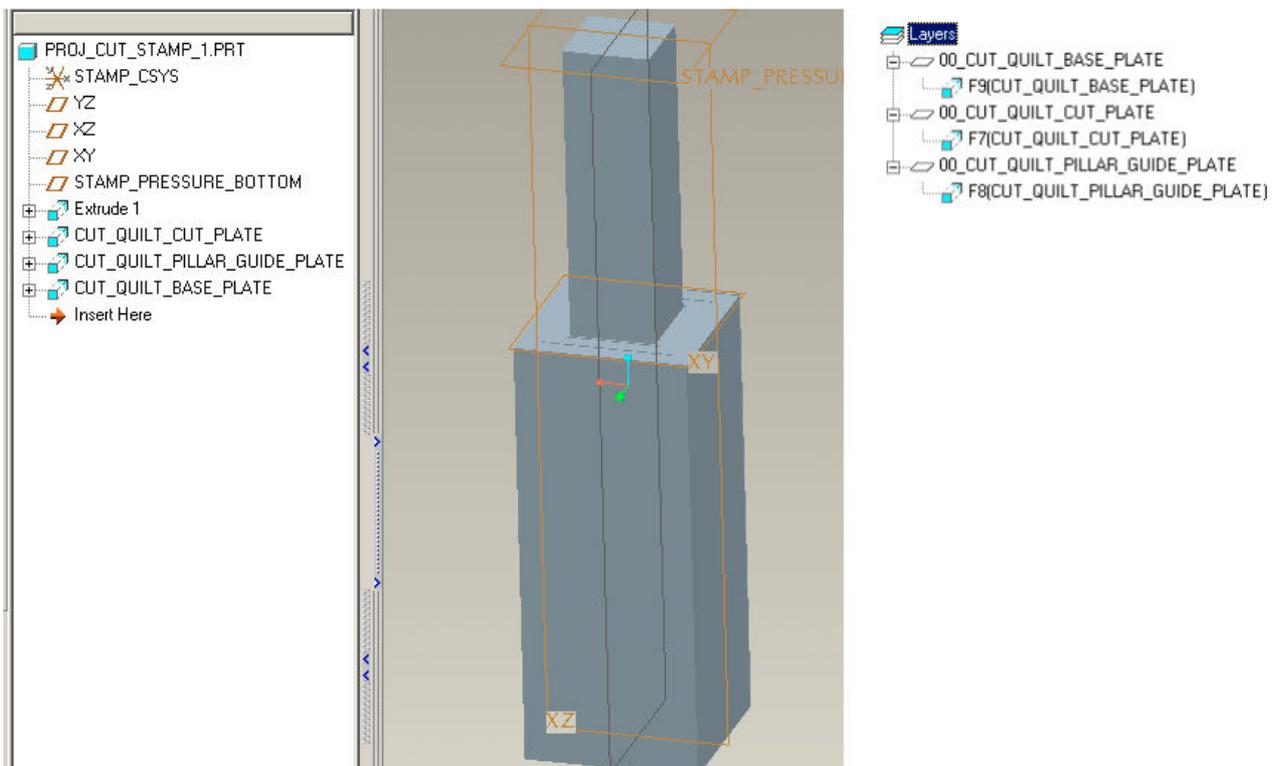
3.6 Cut outs

At the moment the stamp is created, but no cuts are applied to the Die Set. All plates above the strip which are traversed by the stamp should be cut without a gap. In contrast for the plates below cuts with increasing gaps will be created. This should guarantee that no material clamps and causes damage to the machine.

The tutorial will start with the cuts below the strip. In detail the cut plate, pillar guide plate and base plate should be cut with increasing shapes. The clip geometry is determined by different surfaces belonging to the cut stamp. A collision detection checks if a cut is required or not. Only displayed surfaces are considered by the collision detection, therefore each surface is placed on a single layer to switch them on and off for testing.

Exercise:

- Add three extruded surfaces with capped ends, named *CUT_QUILT_CUT_PLATE*, *CUT_QUILT_PILLAR_GUIDE_PLATE* and *CUT_QUILT_BASE_PLATE*. Place the sketch on the XY plane, use 100 in length and 1, 4, and 10 as offset in the sketch.
- Create the layers *00_CUT_QUILT_BASE_PLATE*, *00_CUT_QUILT_CUT_PLATE*, *00_CUT_QUILT_PILLAR_GUIDE_PLATE*, place the surfaces on these and hide them.



Also the component description file for the cut stamp has to be updated.

Exercise:

- The lines marked in bold should to be appended to graphical user description:

```
GLOBAL_PICTURE cut_stamp_tab.gif

USER_SELECT POINT POINT 1
USER_SELECT FACE STAMP_PRESSURE_PLANE 2

USER_SELECT FACE CUT_PLANE 3
USER_SELECT FACE PILLAR_GUIDE_PLANE 4
USER_SELECT FACE BASE_PLANE 5

USER_INPUT_PARAM DOUBLE STAMP_WIDTH
USER_INPUT_PARAM DOUBLE STAMP_DEPTH
USER_INPUT_PARAM DOUBLE STAMP_LENGTH

SHOW_PARAM DOUBLE MIN_LENGTH

SEARCH_STAMP_REF POINT BOTTOM STAMP_PRESSURE_PLATE STAMP_PRESSURE_PLANE
SEARCH_STAMP_REF POINT BOTTOM CUT_PLATE CUT_PLANE
SEARCH_STAMP_REF POINT BOTTOM PILLAR_GUIDE_PLATE PILLAR_GUIDE_PLANE
SEARCH_STAMP_REF POINT BOTTOM BASE_PLATE BASE_PLANE

MEASURE_DISTANCE POINT STAMP_PRESSURE_PLANE MIN_LENGTH
```

Information:

- The new lines are searching for bottom surfaces of cut plate, pillar guide plate and base plate. Because of the *USER_SELECT* lines, the user is able to overwrite the detected surfaces with other references. Also it provides information whether the references are found or not.

Exercise:

- The lines marked in bold should be appended to assembling description:

```
SEARCH_MDL_REF ASSEMBLY PLANE A_XZ A_XZ
SEARCH_MDL_REF ASSEMBLY PLANE A_YZ A_YZ

SEARCH_MDL_REF THIS QUILT CUT_QUILT_CUT_PLATE CUT_QUILT_CUT_PLATE
SEARCH_MDL_REF THIS QUILT CUT_QUILT_PILLAR_GUIDE_PLATE CUT_QUILT_PILLAR_GUIDE_PLATE
SEARCH_MDL_REF THIS QUILT CUT_QUILT_BASE_PLATE CUT_QUILT_BASE_PLATE

CREATE_UDF STAMP_REF ASSEMBLY
  UDF_REF REF_POINT POINT
  UDF_REF REF_PLANE_XZ A_XZ
  UDF_REF REF_PLANE_YZ A_YZ
  UDF_EXP_REF REF_CSYS CSYS 0
END_CREATE_UDF

CREATE_UDF MEASURE_DIST_POINT_FACE ASSEMBLY MEASURE_UDF
  UDF_REF REF_POINT POINT
  UDF_REF REF_FACE STAMP_PRESSURE_PLANE
END_CREATE_UDF

ASSEMBLE THIS_MEASURECOPY
  MEASURE_UDF DISTANCE
  CSYS REF_CSYS STAMP_CSYS
END_ASSEMBLE

SET_LAYER_STATUS THIS 00_CUT_QUILT_CUT_PLATE DISPLAY

IF INTERF CUT_QUILT_CUT_PLATE CUT_PLANE
  CREATE_UDF QUILT_CUT_CUT_PLANE
  UDF_REF REF_QUILT CUT_QUILT_CUT_PLATE
  END_CREATE_UDF
END_IF

SET_LAYER_STATUS THIS 00_CUT_QUILT_CUT_PLATE BLANK
```

```

SET_LAYER_STATUS THIS 00_CUT_QUILT_PILLAR_GUIDE_PLATE DISPLAY

IF INTERF CUT_QUILT_PILLAR_GUIDE_PLATE PILLAR_GUIDE_PLANE
    CREATE_UDF QUILT_CUT PILLAR_GUIDE_PLANE
        UDF_REF REF_QUILT CUT_QUILT_PILLAR_GUIDE_PLATE
    END_CREATE_UDF
END_IF

SET_LAYER_STATUS THIS 00_CUT_QUILT_PILLAR_GUIDE_PLATE BLANK

SET_LAYER_STATUS THIS 00_CUT_QUILT_BASE_PLATE DISPLAY

IF INTERF CUT_QUILT_BASE_PLATE BASE_PLANE
    CREATE_UDF QUILT_CUT BASE_PLANE
        UDF_REF REF_QUILT CUT_QUILT_BASE_PLATE
    END_CREATE_UDF
END_IF

SET_LAYER_STATUS THIS 00_CUT_QUILT_BASE_PLATE BLANK

```

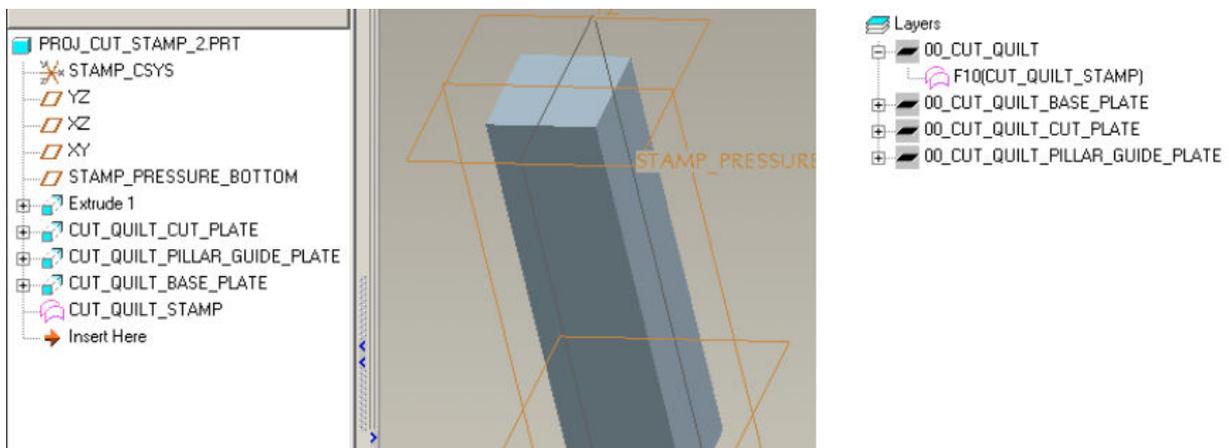
Information:

- The required quilt surfaces for cutting are searched in the component via *SEARCH_MDL_REF*.
- Each layer of a surface is displayed.
- It is tested if the *cut_stamp* interferes with a specific plate.
- In case of interference the cut is applied in this plate as UDF *QUILT_CUT*. This UDF is supplied by the system UDFs in <PDX>\mm\parts\udf.
- The layer is blanked again.

At the moment the cuts above the strip are missing. These should use the exact stamp geometry without any gap.

Exercise:

- Create a surface copy of all solid surfaces named *CUT_QUILT_STAMP*.
- Place it on a layer called *00_CUT_QUILT*
- Blank the layer



Following the assembling description has to be updated to figure out all interfered plates and to apply a cut to them.

Exercise:

- Add the following lines to assembling description:

```
SEARCH_MDL_REF THIS QUILT CUT_QUILT_STAMP CUT_QUILT_STAMP

FOR CUTTED_PLATE REF INTERF_MDL THIS
  CREATE_UDF QUILT_CUT CUTTED_PLATE
    UDF_REF REF_QUILT CUT_QUILT_STAMP
  END_CREATE_UDF
END_FOR
```

Information:

- The *FOR ... END_FOR* embraces all references which are found by the intersection detection.
- In each loop the variable *CUTTED_PLATE* contains the next intersection.
- Displaying the layer is not required since the intersection is caused by stamp geometry.

3.7 BOM

When checking the Pro/ENGINEER report file after assembling this component no stamp will be listed.

Based on the standard values of param_relations.txt the bill of material information should be appended. The following relations are added to each component

```
PDX_PART_NAME = BUW_NAME
PDX_ORD_NUMBER = BUW_TYPE + BUW_SIZE
```

As part name "Sample cut stamp " should be used, and the ordering number should be named e.g. "Stamp 20x20x80" according to width, depth and length of the stamp. The parameters *BUW_NAME*, *BUW_TYPE* and *BUW_SIZE* are assumed as string parameters, therefor the dimension values have to be converted via Pro/ENGINEER relation function *itos*.

Exercise:

- Append the following string parameters on part level

BUW_NAME	with content	Sample cut Stamp
BUW_TYPE	with content	Stamp

- Append the following relation on part level

```
BUW_SIZE = itos(WIDTH) + " X " + itos(DEPTH) + " x " + itos(LENGTH)
```

After assembling a new cut stamp the correct values should be filled in the Pro/ENGINEER report.

3.8 Assembling existent component

PDX provides a mechanism to assemble existing components. This functionality requires some further modifications in the component description file. The interface should not be executed, but some values have to be requested from the user and some values need to be stored and retrieved from the component.

To distinguish the situation in the component description file, the conditions *ELEM_NEW* and *ELEM_EXIST* could be tested via an IF statement.

Exercise:

- Add the modifications marked in bold to the assembling description.

```
SEARCH_MDL_PARAM THIS WIDTH STAMP_WIDTH
SEARCH_MDL_PARAM THIS DEPTH STAMP_DEPTH
SEARCH_MDL_PARAM THIS LENGTH STAMP_LENGTH

SEARCH_MDL_REF THIS CSYS STAMP_CSYS STAMP_CSYS

IF ELEM_NEW
    CONFIG_ELEM
END_IF

IF ELEM_EXIST
    USER_SELECT POINT POINT

    RETRIEVE REFERENCE COMP_FEAT CUT_PLANE
    RETRIEVE REFERENCE COMP_FEAT PILLAR_GUIDE_PLANE
    RETRIEVE REFERENCE COMP_FEAT BASE_PLANE
END_IF

SEARCH_MDL_REF ASSEMBLY PLANE A_XZ A_XZ
SEARCH_MDL_REF ASSEMBLY PLANE A_YZ A_YZ

SEARCH_MDL_REF THIS QUILT CUT_QUILT_CUT_PLATE CUT_QUILT_CUT_PLATE
SEARCH_MDL_REF THIS QUILT CUT_QUILT_PILLAR_GUIDE_PLATE CUT_QUILT_PILLAR_GUIDE_PLATE
SEARCH_MDL_REF THIS QUILT CUT_QUILT_BASE_PLATE CUT_QUILT_BASE_PLATE

CREATE_UDF STAMP_REF ASSEMBLY
    UDF_REF REF_POINT POINT
    UDF_REF REF_PLANE_XZ A_XZ
    UDF_REF REF_PLANE_YZ A_YZ
    UDF_EXP_REF REF_CSYS CSYS 0
END_CREATE_UDF

IF ELEM_NEW
    CREATE_UDF MEASURE_DIST_POINT_FACE ASSEMBLY MEASURE_UDF
        UDF_REF REF_POINT POINT
        UDF_REF REF_FACE STAMP_PRESSURE_PLANE
    END_CREATE_UDF
END_IF

ASSEMBLE THIS_MEASURECOPY
    MEASURE_UDF DISTANCE
    CSYS REF_CSYS STAMP_CSYS
END_ASSEMBLE

SAVE REFERENCE COMP_FEAT CUT_PLANE
SAVE REFERENCE COMP_FEAT PILLAR_GUIDE_PLANE
SAVE REFERENCE COMP_FEAT BASE_PLANE
```

Information:

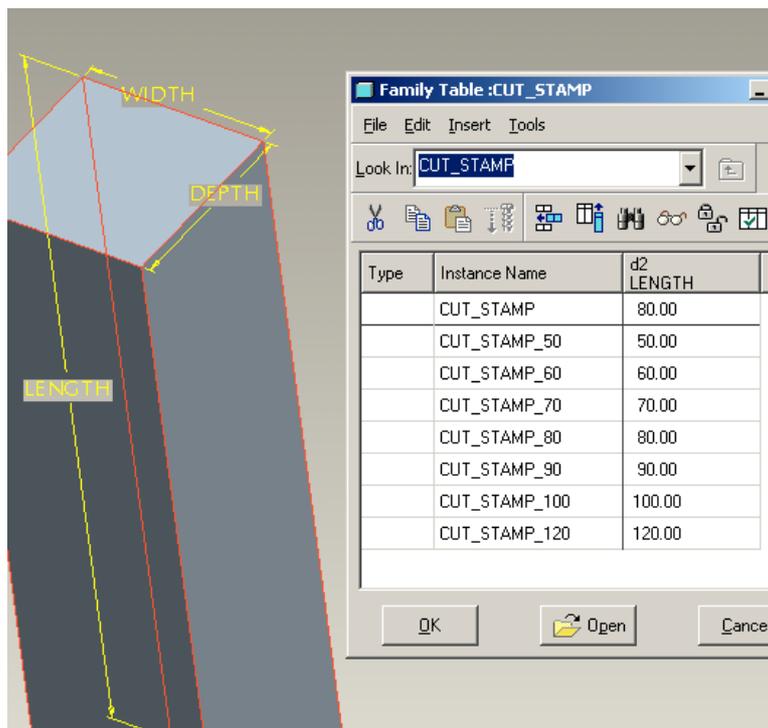
- *IF ELEM_NEW ... END_IF* encapsulates the *CONFIG_ELEM* command which executes the graphical user interface. Therefore it will only show up in case of assembling a new component.
- *IF ELEM_EXIST ... END_IF* encapsulates the selection of a datum point and the retrieval of information about the required cut, pillar guide and base planes.
- *Creation of analysis feature is only created in case of creating a new component*
- *References for cut, pillar guide and base plane will be saved to the stamp.*

4 Use family table instances

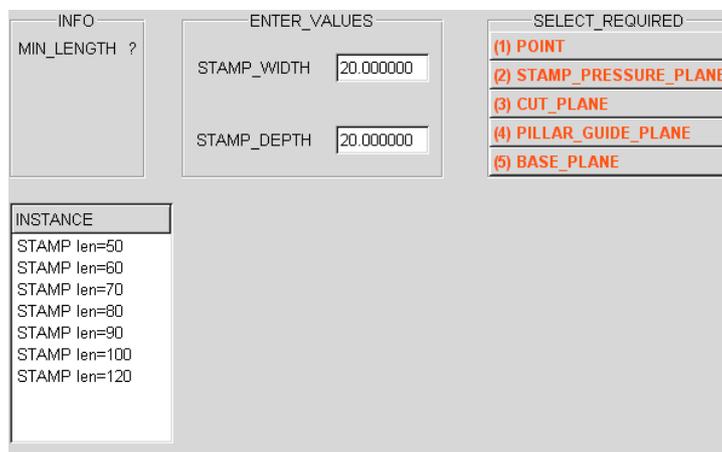
PDX customization also includes the use of family table instances which could be selected by graphical user interface. In the stamp example the length should not longer be entered as arbitrary value. Several lengths should be offered in the interface.

Exercise:

- Create a family table on the stamps according instance names and length values in the following figure.



Now the graphical user interface needs to be modified. Instead of offering a field to enter dimension values, a table for selecting the instance should be supplied like shown in the following figure:



Exercise:

- Append the following table description block to the component description file and remove the *USER_INPUT_PARAM* in for the *STAMP_LENGTH* in GUI description. Since values with blanks are allowed, the columns need to be separated by tabulators (symbolized as “→”).

```
BEGIN_TAB_DESCR

BEGIN_TABLE→STAMP→INSTANCE
SEL_STRING→MDL_NAME
STRING→STRING
STAMP len=50→CUT_STAMP_50<CUT_STAMP>
STAMP len=60→CUT_STAMP_60<CUT_STAMP>
STAMP len=70→CUT_STAMP_70<CUT_STAMP>
STAMP len=80→CUT_STAMP_80<CUT_STAMP>
STAMP len=90→CUT_STAMP_90<CUT_STAMP>
STAMP len=100→CUT_STAMP_100<CUT_STAMP>
STAMP len=120→CUT_STAMP_120<CUT_STAMP>
END_TABLE

END_TAB_DESCR
```

Information:

- Each table is embedded in *BEGIN_TABLE* ... *END_TABLE* statements.
- In first line after *BEGIN_TABLE* statement, table name and display name are following.
- Second line contains the variable which will be defined when selecting a table line. The first column always contains *SEL_STRING* since this is the name that is displayed for each table line. In this example the second column *MDL_NAME* has a special meaning. It determines the naming of the component. This only works for *part* or *asm* settings in *sel_list.txt*, *part_copy* and *asm_copy* will generate new names for each component.
- Third line contains the data types of the table. This example is only using strings.
- Then the data follows. First column contains the string which is displayed in the table (e.g. *STAMP len=50*). In the second row the name for the component is given. Pro/ENGINEER supports specification for family table instances by *INSTANCE_NAME<GENERIC_NAME>*. As an example *CUT_STAMP_50<CUT_STAMP>* means instance *CUT_STAMP_50* from generic *CUT_STAMP*.

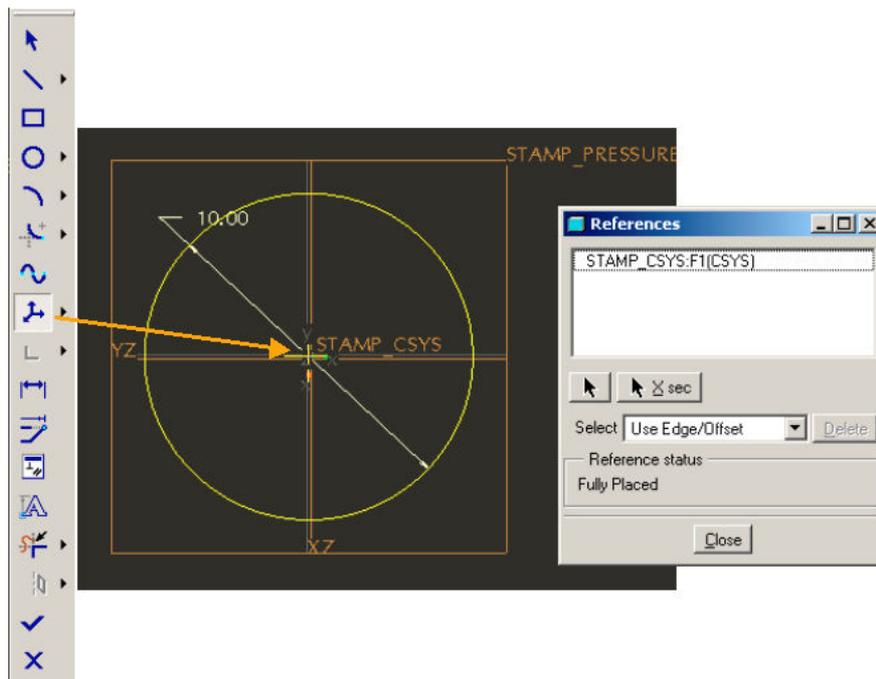
Exercise:

- modify *sel_list.txt* entry from *part_copy* to *part* and test the component.

5 Contoured cut stamps

In some situations the contour of the cut stamps should follow a specific shape. Therefore stamp reference parts in the strip are provided to define this shape. After being assembled stamp geometry and cut geometry should be adopted. The component engine provides a special command to redefine a sketched feature to a plane surface shape.

Some preconditions for features which are redefined have to be considered. Only a reference to a coordinate system should be supplied. The first element on the section must be a sketching coordinate system which is located on the reference coordinate system. All other sketched elements only should depend on this coordinate system and on dimensions. It is recommended to use circles as shapes for feature which are to be redefined.



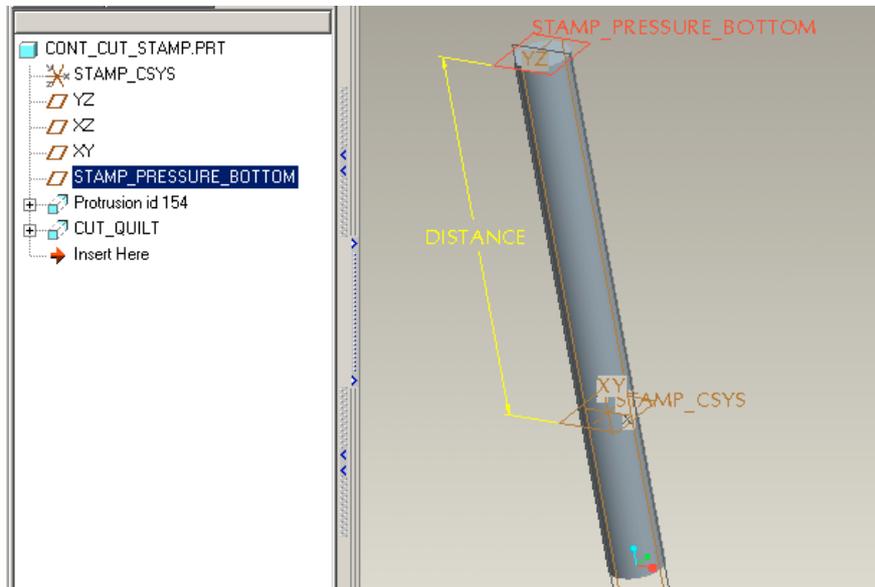
The first step is to create a new Pro/ENGINEER part for the contoured cut stamp.

Exercise:

- Create Pro/ENGINEER part called *cont_cut_stamp.prt* in folder <PDX>\mm\component_engine\cut_stamps\sample.
- Create standard coordinate system *STAMP_CSYS* and standard datum planes *YZ*, *YZ*, *XY* as in the cut stamp example in chapter 3.
- Create offset datum plane *STAMP_PRESSURE_BOTTOM* with offset 70. Set symbolic name to *DISTANCE* as in cut stamp example in chapter 3.
- Create a round solid protrusion feature sketched on *STAMP_PRESSURE_BOTTOM*. This feature has to be redefined later on, therefore make sure to follow the guideline above. Set protrusion length to 100 and modify the symbolic name to *LENGTH*.

- Create a round surface feature *CUT_QUILT* similar to the protrusion feature before. Set length to 200. This feature has to be redefined later on, therefore make sure to follow the guideline above.
- Create a layer called *00_CUT_QUILT*, place feature *CUT_QUILT* on it and hide the layer.

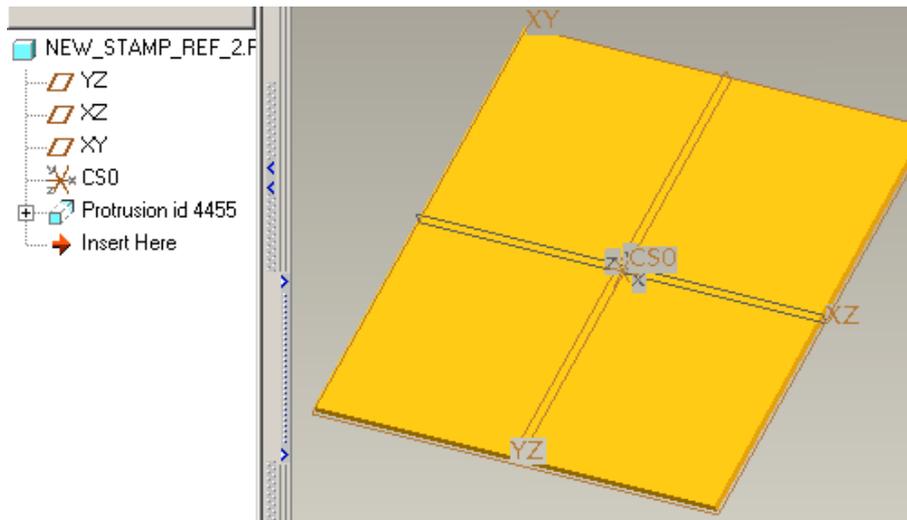
Now the stamp should look like the following figure.



- Create a small gif image with height 100 of the stamp and save it as *cont_cut_stamp.gif*. Add also a larger gif image *cont_cut_stamp_tab.gif*. Both should be located in the sample folder.
- A second line in *sel_list.txt* from sample folder has to be added.

```
cont_cut_stamp part_copy
```

Similar to cut stamp of chapter 3 the stamp should be located on strip level, but it needs to be ensured, that it is aligned to stamp pressure plate. Handling the component only the top surface of a stamp reference part should be selected.



The coordinate system of the strip *CS0* will be used as source for an UDF to create assembling constraints.

Exercise:

- Create the following component description `cont_cut_stamp.tab` for the contoured cut stamp:

```

BEGIN_GUI_DESCR

GLOBAL_PICTURE cont_cut_stamp_tab.gif

USER_SELECT_FACE STAMP_REF_TOP 1
USER_INPUT_PARAM DOUBLE STAMP_LENGTH

SHOW_PARAM DOUBLE MIN_LENGTH

SEARCH_STAMP_REF STAMP_REF_TOP TOP STAMP_HEAD_PLATE STAMP_HEAD_PLANE

MEASURE_DISTANCE STAMP_REF_TOP STAMP_HEAD_PLANE MIN_LENGTH

END_GUI_DESCR

BEGIN_ASM_DESCR

SEARCH_MDL_PARAM THIS LENGTH STAMP_LENGTH

CONFIG_ELEM

SEARCH_MDL_REF THIS CSYS STAMP_CSYS STAMP_CSYS
SEARCH_MDL_REF STAMP_REF_TOP PLANE XY STAMP_REF_XY
SEARCH_MDL_REF STAMP_REF_TOP CSYS CS0 REF_CSYS

CREATE_UDF CSYS_XY_OFF_Z_ROT ASSEMBLY CENTER_CSYS
  UDF_REF REF_CSYS REF_CSYS
  UDF_DIM X_OFF 0
  UDF_DIM Y_OFF 0
  UDF_DIM Z_ROT 0
  UDF_EXP_REF ASM_CSYS CSYS 0
END_CREATE_UDF

CREATE_UDF PLANE_DIST_MEASURE ASSEMBLY MEASURE_UDF
  UDF_REF PLANE1 STAMP_REF_XY
  UDF_REF PLANE2 STAMP_HEAD_PLANE
END_CREATE_UDF

ASSEMBLE THIS_MEASURECOPY
  MEASURE_UDF DISTANCE
  CSYS ASM_CSYS STAMP_CSYS
END_ASSEMBLE

```

```

SEARCH_MDL_REF THIS QUILT CUT_QUILT CUT_QUILT

SET_LAYER_STATUS THIS 00_CUT_QUILT DISPLAY

FOR CUT_MDL REF INTERF_MDL THIS
  CREATE_UDF QUILT_CUT CUT_MDL
    UDF_REF REF_QUILT CUT_QUILT
  END_CREATE_UDF

END_FOR

SET_LAYER_STATUS THIS 00_CUT_QUILT BLANK

END_ASM_DESCR

```

Information:

- The UDFs *CSYS_XY_OFF_Z_ROT* and *PLANE_DIST_MEASURE_ASSEMBLY* are already available in <PDX>\mm\parts\udf.
- The *SEARCH_MDL_REF* line marked in bold specifies the component in which to search for a plane by supplying an element of this component (selected surface of stamp ref part).

Using the component right now will not lead to the desired result. The component will be placed correctly, but the shape will not adopt. This behavior could be obtained by the *LINK_SKETCH* command. It specifies the feature to be redefined by its feature id.

	Feat ID
CONT_CUT_STAMP.PR1	
✂ STAMP_CSYS	1
YZ	3
XZ	5
XY	7
STAMP_PRESSURE	36
Protrusion id 154	154
CUT_QUILT	245
Insert Here	

Exercise:

- Add the following lines after the stamp is assembled:

```

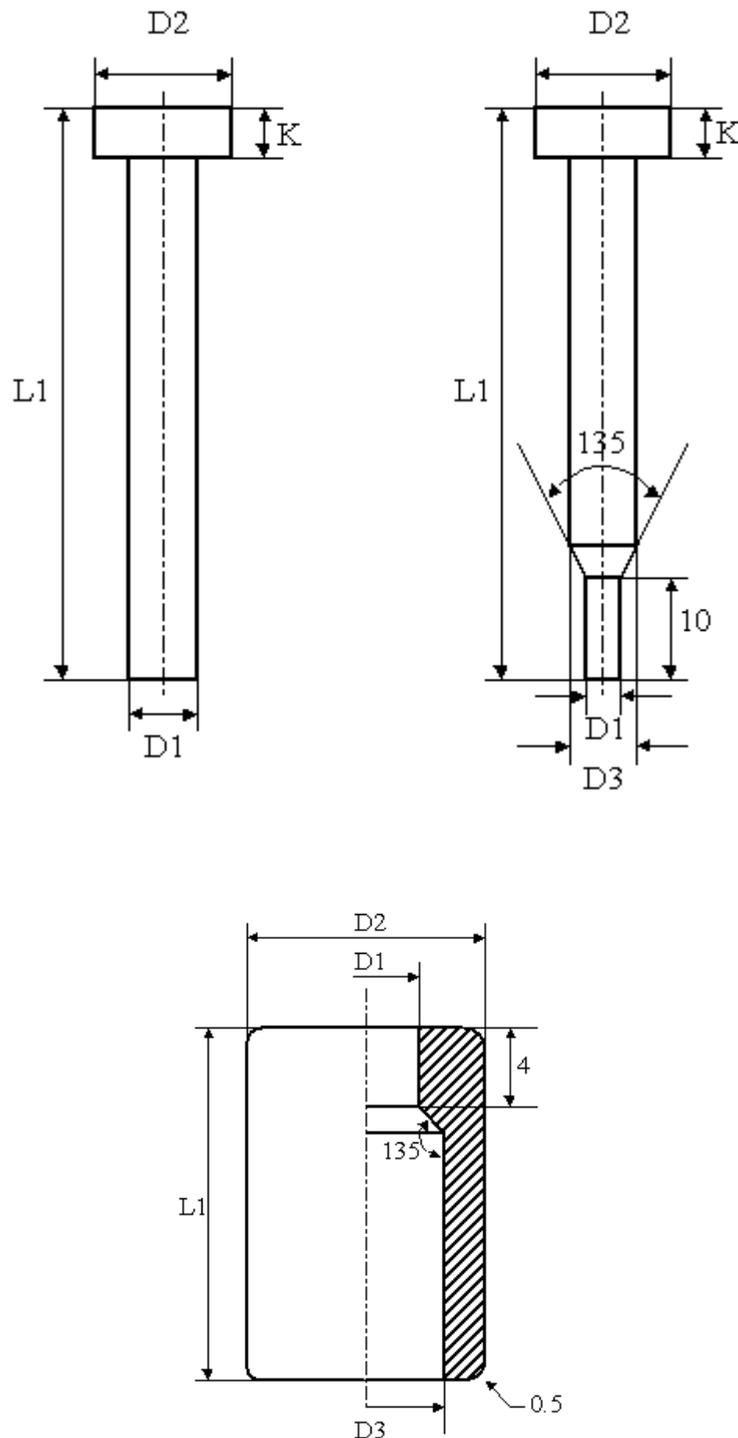
LINK_SKETCH THIS 154 STAMP_REF_TOP
LINK_SKETCH THIS 245 STAMP_REF_TOP

```

6 Creating a more complex cut stamp as assembly

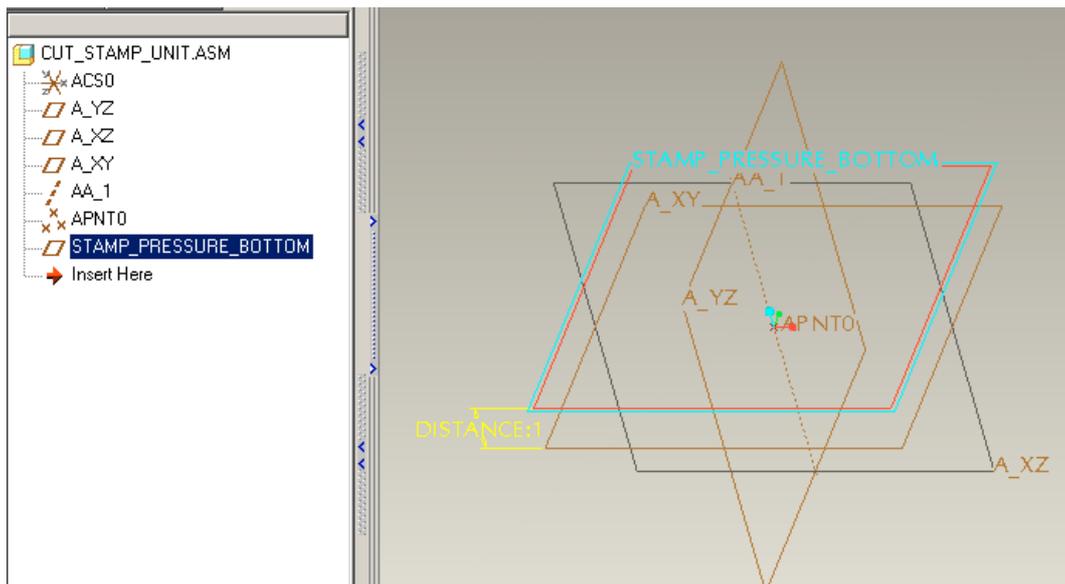
All examples up to this state consists of placement of one single part which could be modified by changing dimension values. This chapter shows how to setup a more complex assembly component on the example of a cut stamp with cut bush. This stamping unit offers two different stamp types – a straight stamp and a stepped stamp. Below the strip either a bush should be supplied or a through hole.

6.1 Creating and assembling cut stamp unit

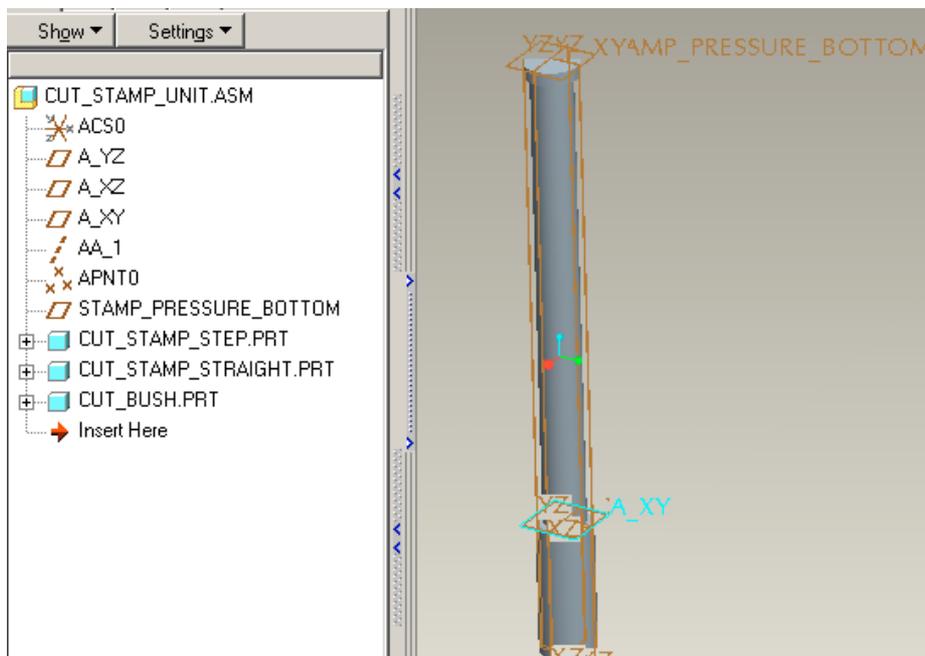


Exercise:

- Create the stamps and the bush shown in the figure above. Save them as *cut_stamp_straight*, *cut_stamp_step* and *cut_bush* to folder <PDX>mm\component_engine\cut_stamps\sample.
- Create cut_stamp_unit with datum planes A_YZ, A_XZ, A_XY like shown in the figure below. Create axis AA_1 created by planes A_XZ and A_YZ. Model a datum point APNT0 by axis AA_1 and plane A_XY and build an offset datum plane STAMP_PRESSURE_PLANE with offset DISTANCE and value 70.



- Now stamps and cut bush should be assembled. The stamp heads must be aligned with STAMP_PRESSURE_BOTTOM and axis AA_1. Cut bush has to be aligned with datum plane A_XY and also with axis AA_1.



- Add the following line to sel_list.txt in <PDX>mm\component_engine\cut_stamps\sample:
cut_stamp_unit asm_copy
- Create gif images for the component choice and for the component dialog named *cut_stamp_unit.gif* and *cut_stamp_unit_tab.gif*.

The next step is to determine component placement. In this example the user should specify a datum point on the lower side of the strip to specify the position. This datum point will be aligned with datum point *APTNO* from the component. As further reference the plane *A_XY* from component should be oriented like plane *A_XY* from project assembly. Distance between *STAMP_PRESSURE_PLANE* and *A_XY* will be controlled by an analysis feature.

Exercise:

- Create the following component description file named *cut_stamp_unit.tab* and save it in folder <PDX>mm\component_engine\cut_stamps\sample.

```

BEGIN_GUI_DESCR

GLOBAL_PICTURE cut_stamp_unit_tab.gif

USER_SELECT POINT PLACEMENT_POINT 1
USER_SELECT PLANE STAMP_PRESSURE_PLANE_BOTTOM 2

SHOW_PARAM DOUBLE MIN_LENGTH

SEARCH_STAMP_REF PLACEMENT_POINT BOTTOM STAMP_PRESSURE_PLATE STAMP_PRESSURE_PLANE
MEASURE_DISTANCE PLACEMENT_POINT STAMP_PRESSURE_PLANE MIN_LENGTH

END_GUI_DESCR

BEGIN_ASM_DESCR

SEARCH_MDL_REF THIS POINT APNT0 STAMP_POINT
SEARCH_MDL_REF THIS PLANE A_XY STAMP_PLANE_XY
SEARCH_MDL_REF ASSEMBLY PLANE A_XY TOOL_PLANE_XY

IF ELEM_NEW
    CONFIG_ELEM
END_IF

IF ELEM_EXIST
    USER_SELECT POINT PLACEMENT_POINT
END_IF

IF ELEM_NEW
    CREATE_UDF MEASURE_DIST_POINT_FACE ASSEMBLY MEASURE_UDF
        UDF_REF REF_POINT PLACEMENT_POINT
        UDF_REF REF_FACE STAMP_PRESSURE_PLANE
    END_CREATE_UDF
END_IF

ASSEMBLE THIS_MEASURECOPY
    MEASURE_UDF DISTANCE
    ALIGN PLACEMENT_POINT STAMP_POINT
    ORIENT TOOL_PLANE_XY STAMP_PLANE_XY
END_ASSEMBLE

END_ASM_DESCR

```

6.2 Component configuration

Since the cut stamp unit contains all stamps and bushes that could be configured by the user, a mechanism is required to determine which are the required ones. As in chapter 4 tables will be used to provide a selection mechanism for the user. In this example more than one table is required for selection. Therefore a mechanism is introduced how to handle sub tables.

Additional it is required to remove all sub components which are not required according to the configuration.

Exercise:

- Add the following table description block to *cut_stamp_unit.tab*. Tabulators are symbolized as “ → ” in the following code:

```
BEGIN_TAB_DESCR

BEGIN_TABLE→STAMP→STAMP
SEL_STRING→STAMP_TABLE→CUT_BUSH_TABLE
STRING→SUBTABLE→SUBTABLE
XXX→STAMP_TABLE→CUT_BUSH_TABLE
END_TABLE

BEGIN_TABLE→STAMP_TABLE→Stamps
SEL_STRING→DEL_IDS_STAMPS→STAMP_TYPE
STRING→CONFIG_DELETE_IDS→INTEGER
Cut stamp straight→19→1
Cut stamp stepped→23→2
END_TABLE

BEGIN_TABLE→CUT_BUSH_TABLE→Cut Bush
SEL_STRING→DEL_IDS_BUSH→BUSH_TYPE
STRING→CONFIG_DELETE_IDS→INTEGER
Cut bush→NO_VALUE→1
Through hole→24→2
END_TABLE

END_TAB_DESCR
```

Information:

- The system starts to execute the very first table. In that example this is the table *STAMP*. In the line which contains the data type (third line of each table), *SUBTABLE* is used as specification. This means that evaluation of a further table is required. In this case *STAMP_TABLE* and *CUT_BUSH_TABLE* have to be evaluated.
- The table *STAMP* contains only one line of data – the specification of other tables which have to be evaluated. Therefore the system could automatically determine which is the right choice and the table will not show up in interface.
- The tables *STAMP_TABLE* and *CUT_BUSH_TABLE* contain the data type *CONFIG_DELETE_IDS*. This specifies that a sub component will be removed according this table selection. For example when user selects *Cut stamp straight* in table *STAMP_TABLE* component with id 19 is removed. As shown in the figure below this is the *cut_stamp_step.prt.NO_VALUE*, used in table *CUT_BUSH_TABLE*, means that nothing has to be removed.

	Feat ID
CUT_STAMP_UNIT.ASM	
CUT_STAMP_STEP.PRT	19
CUT_STAMP_STRAIGHT.PRT	23
CUT_BUSH.PRT	24

6.3 Define component dimensions

After the user has selected the configuration of the cut stamp unit, dimensional values have to be provided. Since these values typically follow supplier catalogs, they should also be provided via selection tables. An example table for straight and stepped cut stamp as well as for the bushing follows:

cut stamp straight

L1	D1	D2	K
70	3	5	3
80			
70	4	6	3
80			
70	5	8	5
80			
70	6	9	5
80			
70	7	11	5
80			

cut stamp stepped

L1	D1	D2	D3	K
70	1	5	3	3
80				
70	1.5	6	4	3
80				
70	2	8	5	5
80				
70	2.5	9	6	5
80				
70	3	11	8	5
80				

cut bush

L1	D1	D2	D3
20	1	5	2.8
25			
20	1.5	5	2.8
25			
20	2	6	3.5
25			
20	2.5	6	3.5
25			
20	3	8	4
25			

L1	D1	D2	D3
20	3.5	8	4
25			
20	4	10	5.8
25			
20	5	10	5.8
25			
20	6	13	8
25			
20	8	16	9.5
25			

Since only stamps should be dimensioned which are selected, the mechanism will use a component description.

Exercise:

- Append the column marked in bold to table description of stamp and bush table in *cut_stamp_unit.tab*.

```
BEGIN_TABLE→STAMP_TABLE→Stamps
SEL_STRING→STAMP_SUBCOMP→DEL_IDS_STAMPS→STAMP_TYPE
STRING→SUBCOMP→CONFIG_DELETE_IDS→INTEGER
Cut stamp straight→STRAIGHT_STAMP→19→1
Cut stamp stepped→STEPPED_STAMP→23→2
END_TABLE
```

```
BEGIN_TABLE→CUT_BUSH_TABLE→Cut Bush
SEL_STRING→BUSH_SUBCOMP→DEL_IDS_BUSH→BUSH_TYPE
STRING→SUBCOMP→CONFIG_DELETE_IDS→INTEGER
Cut bush→CUT_BUSH→NO_VALUE→1
Through hole→NO_VALUE→24→2
END_TABLE
```

- Append the component description block

```
BEGIN_COMP_DESCR

BEGIN_SUBCOMP→STRAIGHT_STAMP→23
CONFIG_SUBCOMP→CUT_STAMP_STR
END_SUBCOMP

BEGIN_SUBCOMP→STEPPED_STAMP→19
CONFIG_SUBCOMP→CUT_STAMP_STEP
END_SUBCOMP

BEGIN_SUBCOMP→CUT_BUSH→24
CONFIG_SUBCOMP→CUT_BUSH
END_SUBCOMP

END_COMP_DESCR
```

Information:

- In table description block sub components are defined. When user selects *Cut stamp straight*, the sub component *STRAIGHT_STAMP* of component description block will be executed.
- In the component description block three sub components are configured. E.g. *STRAIGHT_STAMP* configures the component with id 23 and executes component description file *cut_stamp_str.tab* for this.

Now the three component description files for sub component configuration need to be defined and gif images should be created for each of them.

Exercise:

- Create gif images *cut_stamp_step_tab.gif*, *cut_stamp_str_tab.gif* and *cut_bush_tab.gif* and save them to folder <PDX>mm\component_engine\cut_stamps\sample.
- Create *cut_stamp_str.tab*, *cut_stamp_step.tab* and *cut_bush.tab* in this folder with the following content:

cut_stamp_str:

```
BEGIN_GUI_DESCR
GLOBAL_PICTURE cut_stamp_str_tab.gif
END_GUI_DESCR

BEGIN_TAB_DESCR

BEGIN_TABLE→STAMP_LENGTH→STAMP_LENGTH
SEL_STRING→STAMP_LENGTH→TABLE_DIA→MDL_NAME_1
STRING→DOUBLE→SUBTABLE→STRING
Len=70→70→STAMP_DIA→STAMP_STR_70
Len=80→80→STAMP_DIA→STAMP_STR_80
END_TABLE

BEGIN_TABLE→STAMP_DIA→STAMP_DIA
SEL_STRING→D1→D2→K→MDL_NAME_2
STRING→DOUBLE→DOUBLE→DOUBLE→STRING
Dia=3→3→5→3→x3
Dia=4→4→6→3→x4
Dia=5→5→8→5→x5
Dia=6→6→9→5→x6
Dia=7→7→11→5→x7
END_TABLE

END_TAB_DESCR
```

cut_stamp_step:

```
BEGIN_GUI_DESCR

GLOBAL_PICTURE      cut_stamp_step_tab.gif

END_GUI_DESCR

BEGIN_TAB_DESCR

BEGIN_TABLE→STAMP_LENGTH→STAMP_LENGTH
SEL_STRING→STAMP_LENGTH→TABLE_DIA→MDL_NAME_1
STRING→DOUBLE→SUBTABLE→STRING
Len=70→70→STAMP_DIA→STAMP_STEP_70
Len=80→80→STAMP_DIA→STAMP_STEP_80
END_TABLE

BEGIN_TABLE→STAMP_DIA→STAMP_DIA
SEL_STRING→D1→D2→D3→K→MDL_NAME_2
STRING→DOUBLE→DOUBLE→DOUBLE→DOUBLE→STRING
Dia=1→1→5→3→3→x1
Dia=1.5→1.5→6→4→3→x1_5
Dia=2→2→8→5→5→x2
Dia=2.5→2.5→9→6→5→x2_5
Dia=3→3→11→8→5→x3
END_TABLE

END_TAB_DESCR
```

cut_bush.tab:

```
BEGIN_GUI_DESCR

GLOBAL_PICTURE cut_bush_tab.gif

END_GUI_DESCR

BEGIN_TAB_DESCR

BEGIN_TABLE→BUSH_LENGTH→BUSH_LENGTH
SEL_STRING→L1→TABLE_DIA→MDL_NAME_1
STRING→DOUBLE→SUBTABLE→STRING
Len=20→20→BUSH_DIA→CUT_BUSH_20
Len=25→25→BUSH_DIA→CUT_BUSH_25
END_TABLE

BEGIN_TABLE→BUSH_DIA→BUSH_DIA
SEL_STRING→D1→D2→D3→MDL_NAME_2
STRING→DOUBLE→DOUBLE→DOUBLE→STRING
Dia=1→1→5→2.8→x1
Dia=1.5→1.5→5→2.8→x1_5
Dia=2→2→6→3_5→x2
Dia=2.5→2.5→6→3_5→x2_5
Dia=3→3→8→4→x3
Dia=3.5→3.5→8→4→x3_5
Dia=4→4→10→5.8→x4
Dia=5→5→10→5.8→x5
Dia=6→6→13→8→x6
Dia=8→8→16→9.5→x8
END_TABLE

END_TAB_DESCR
```

Information:

- In *cut_stamp_str.tab* *MDL_NAME_1* and *MDL_NAME_2* are configured by the table description. The contents will be concatenated to build up the naming convention for this sub component. For example when the user selects len=70 and dia=3 the name *STAMP_STR_70x3* will be used for the component.
- All dimensional values are determined by use of the table description.

At the moment the user has to specify the diameter for cut bush. Following this should be regarded as equal to stamping diameter. The component engine provides a mechanism to link the parameters by *INPUT_PARAM* and *OUTPUT_PARAM* statements in the component description block:

Exercise:

- Add the lines marked in bold to component description block:

```

BEGIN_SUBCOMP→STRAIGHT_STAMP→23
CONFIG_SUBCOMP→CUT_STAMP_STR
OUTPUT_PARAM STAMP_DIA→D1
END_SUBCOMP

BEGIN_SUBCOMP→STEPPEDED_STAMP→19
CONFIG_SUBCOMP→CUT_STAMP_STEP
OUTPUT_PARAM STAMP_DIA→D1
END_SUBCOMP

BEGIN_SUBCOMP→CUT_BUSH→24
INPUT_PARAM→STAMP_DIA→D1
CONFIG_SUBCOMP→CUT_BUSH
END_SUBCOMP

```

Information:

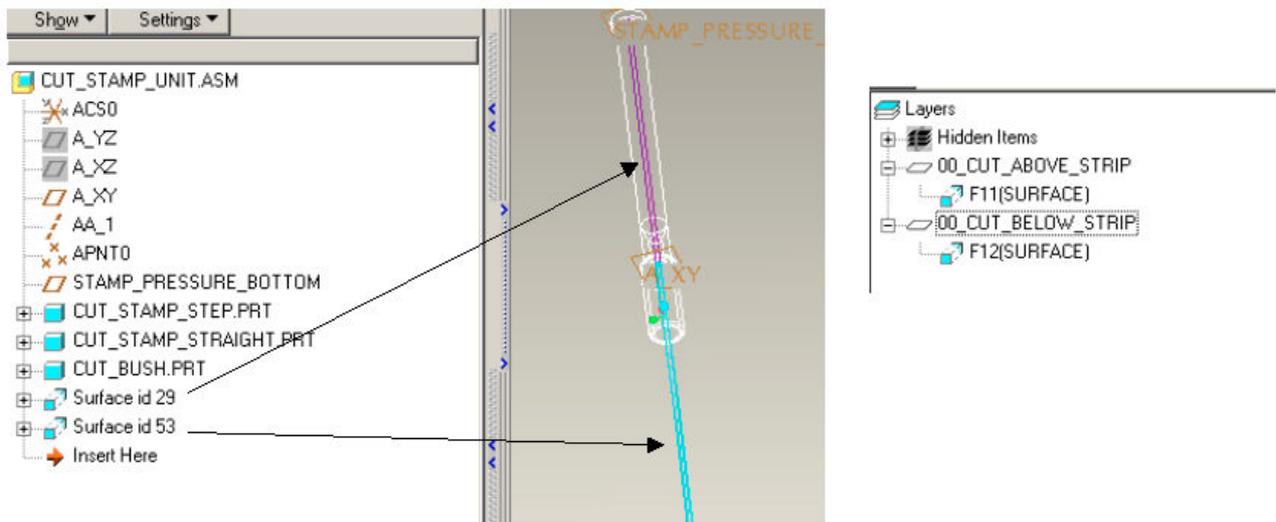
- After processing straight stamp or stepped stamp the *OUTPUT_PARAM* statement assigns value of variable *D1* of *cut_stamp_str.tab* or *cut_stamp_step.tab* to variable *STAMP_DIA*. When user is executing *cut_bush.tab* this value will be assigned to *D1* variable.
- *BUSH_DIA* table from *cut_bush.tab* will now be executed automatically since parameter *D1* is already defined.

6.4 Applying cut outs

In this example the material cuts will not be created by surface quilts, but two surface features are still required to determine intersected plates.

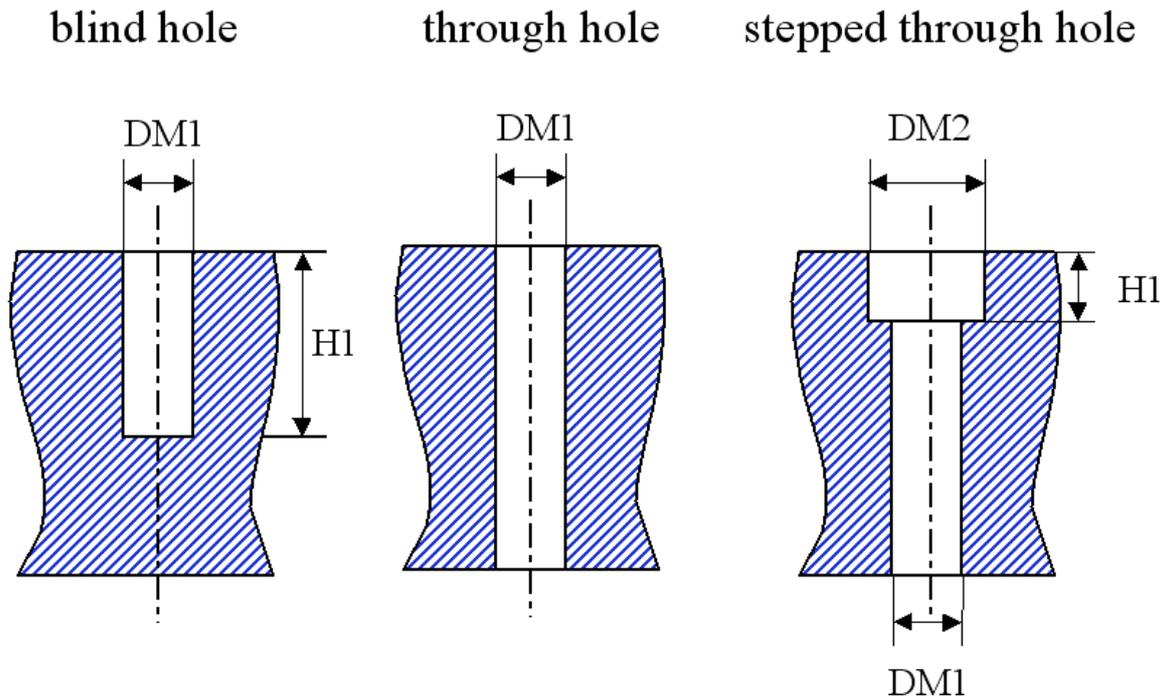
Exercise:

- Create two surface features as protrusion with capped ends as cylindrical shape. First one should be sketched on *STAMP_PRESSURE_BOTTOM* with diameter 1 and depth up to plane *A_XY*. Second should be sketched on *A_XY* with diameter 1 and depth 100.



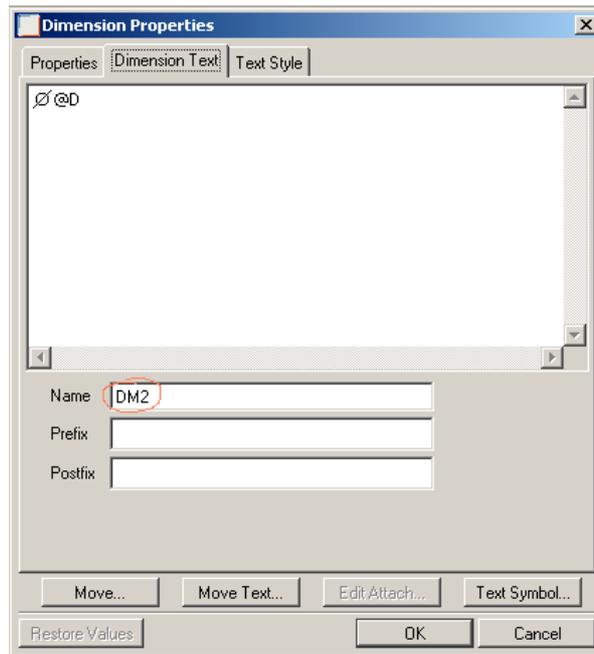
- Place the surfaces on layers – *00_CUT_ABOVE_STRIP* and *00_CUT_BELOW_STRIP* – and blank these.

The example requires the placement of blank holes, through holes and stepped through holes. Therefore these have to be created as UDFs.



Exercise:

- Create the UDFs *blind_hole*, *through_hole* and *through_hole_stepped* according to the figure above.
- Use *STAND ALONE* as *UDF OPTION*. A reference part is not required.
- As placement references a datum point *PLACEMENT_POINT* and a surface *PLACEMENT_SURFACE* should be supplied.
- Apply the variable dimensions from the figure above. Make sure that the dimension name is exactly the same as the variable dimension. Otherwise this will not be considered by the program. You can change the dimension name in Pro/ENGINEER via dimension properties as shown in the figure below:



In the GUI description some further references needs to be asked from the user or determined by the system. The upper side from STAMP_HEAD_PLATE and CUT_PLATE will be required to apply the cuts.

Exercise:

- Add the lines marked in bold to graphical user interface description block.

```

BEGIN_GUI_DESCR

GLOBAL_PICTURE cut_stamp_unit_tab.gif

USER_SELECT POINT PLACEMENT_POINT 1
USER_SELECT PLANE STAMP_PRESSURE_PLANE 2
USER_SELECT PLANE STAMP_HEAD_PLANE 3
USER_SELECT PLANE CUT_PLANE 4

SHOW_PARAM DOUBLE MIN_LENGTH

SEARCH_STAMP_REF PLACEMENT_POINT BOTTOM STAMP_PRESSURE_PLATE STAMP_PRESSURE_PLANE
SEARCH_STAMP_REF PLACEMENT_POINT TOP STAMP_HEAD_PLATE STAMP_HEAD_PLANE
SEARCH_STAMP_REF PLACEMENT_POINT TOP CUT_PLATE CUT_PLANE
MEASURE_DISTANCE PLACEMENT_POINT STAMP_PRESSURE_PLANE MIN_LENGTH

END_GUI_DESCR

BEGIN_TAB_DESCR

```

For applying the right dimensions to the UDFs access to table values from stamp and bush components is required. This could be managed by applying more *OUTPUT_PARAM* statements in the component description block. Additional required dimensions are diameter of stamp head and stamp body as well as height of stamp head.

For straight stamp body diameter is equal to stamp diameter *D1* and for stepped one *D3* is the right choice. In both cases the stamp diameter is *D2* and stamp height is *K*.

Exercise:

- Add the lines marked in bold to component description block:

```

BEGIN_COMP_DESCR

BEGIN_SUBCOMP→STRAIGHT_STAMP→23
CONFIG_SUBCOMP→CUT_STAMP_STR
OUTPUT_PARAM→STAMP_DIA→D1
OUTPUT_PARAM→STAMP_HEAD_DIA→D2
OUTPUT_PARAM→STAMP_BODY_DIA→D1
OUTPUT_PARAM→STAMP_HEAD_HEIGHT→K
END_SUBCOMP

BEGIN_SUBCOMP→STEPPED_STAMP→19
CONFIG_SUBCOMP→CUT_STAMP_STEP
OUTPUT_PARAM→STAMP_DIA→D1
OUTPUT_PARAM→STAMP_HEAD_DIA→D2
OUTPUT_PARAM→STAMP_BODY_DIA→D3
OUTPUT_PARAM→STAMP_HEAD_HEIGHT→K

END_SUBCOMP

BEGIN_SUBCOMP→CUT_BUSH→24
INPUT_PARAM→STAMP_DIA→D1
CONFIG_SUBCOMP→CUT_BUSH
OUTPUT_PARAM→BUSH_DIA→D2
OUTPUT_PARAM→BUSH_LENGTH→L1
END_SUBCOMP

END_COMP_DESCR

```

Now the different UDFs should be placed.

Exercise:

- Add the following lines to assembling description block:

```

IF PARAM_EQUAL BUSH_TYPE 1
    CREATE_UDF BLIND_HOLE CUT_PLANE
        UDF_REF PLACEMENT_POINT PLACEMENT_POINT
        UDF_REF PLACEMENT_SURFACE CUT_PLANE
        UDF_DIM H1 BUSH_LENGTH
        UDF_DIM DM1 BUSH_DIA
    END_CREATE_UDF
END_IF

CREATE_UDF THROUGH_HOLE_STEP STAMP_HEAD_PLANE
    UDF_REF PLACEMENT_POINT PLACEMENT_POINT
    UDF_REF PLACEMENT_SURFACE STAMP_HEAD_PLANE
    UDF_DIM DM1 STAMP_BODY_DIA
    UDF_DIM DM2 STAMP_HEAD_DIA
    UDF_DIM H1 STAMP_HEAD_LENGTH
END_CREATE_UDF

SET_LAYER_STATUS THIS 00_CUT_ABOVE_STRIP DISPLAY

FOR CUTTED_PLATE REF INTERF_MDL THIS
    CREATE_UDF THROUGH_HOLE CUTTED_PLATE
        UDF_REF PLACEMENT_POINT PLACEMENT_POINT
        UDF_REF PLACEMENT_SURFACE STAMP_HEAD_PLANE
        UDF_DIM DM1 STAMP_BODY_DIA
    END_CREATE_UDF
END_FOR

SET_LAYER_STATUS THIS 00_CUT_ABOVE_STRIP BLANK

```

```

SET_LAYER_STATUS THIS 00_CUT_BELOW_STRIP DISPLAY
FOR CUTTED_PLATE REF INTERF_MDL THIS
  CREATE_UDF THROUGH_HOLE CUTTED_PLATE
    UDF_REF PLACEMENT_POINT PLACEMENT_POINT
    UDF_REF PLACEMENT_SURFACE CUT_PLANE
    UDF_DIM          DM1 STAMP_BODY_DIA
  END_CREATE_UDF
END_FOR

SET_LAYER_STATUS THIS 00_CUT_BELOW_STRIP BLANK

```

Information:

- In case a bush should be assembled a blind hole with bush dimensions has to be applied. To distinguish between bush and through hole, the variable *BUSH_TYPE* is used, which is set by table selection.
- In any case the stepped through hole will be applied to do the cut out for stamp head.
- Requirement of further cuts is figured out by collision detection.