UF_SMD_ask_bend_line_data (view source)

Defined in: uf_smd.h

Overview

Returns the bend allowance formula (BAF), bend angle, inside radius, thickness of the solid body, computed bend allowance distance, and inside/outside indicator for the bend line associated of a flat pattern object.

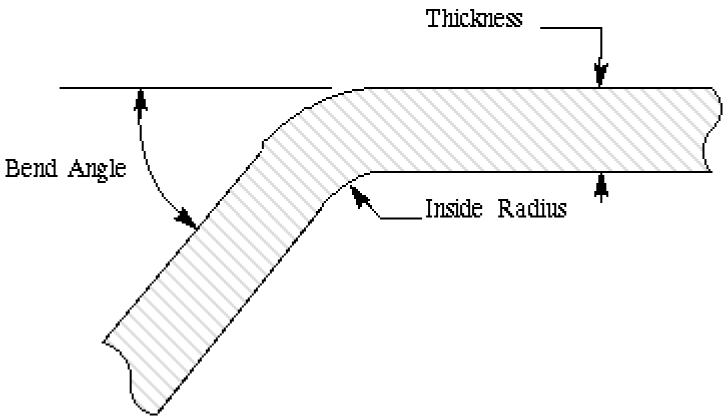

Figure: Illustration of Bend Line Terminology

Environment

Internal and External

Required License(s)

gateway

```
int UF_SMD_ask_bend_line_data
(
    tag_t bend_line,
    char * * bend_formula,
    double * angle,
    double * inside_radius,
    double * thickness,
    double * bend_allowance,
    int * material_side
)
```

tag_t	**bend_line**	Input	Object ID of a bend line (curve).
char * *	**bend_formula**	Output to UF_*free*	Pointer to a character string defining the Bend Allowance Formula. This must be freed by calling UF_free.
double *	**angle**	Output	Bend angle (in Radians).
double *	**inside_radius**	Output	Radius of the bend area.
double *	**thickness**	Output	Thickness of the material

double *	**bend_allowance**	Output	The dimensional amount added to the part through elongation during the bending process derived from the bend allowance formula.
int *	**material_side**	Output	Indicator of the material side for the bend line. 0 = Inside 1 = Outside

UF_SMD_ask_bend_lines (view source)

Defined in: uf_smd.h

Overview

Acquires an array of bend line object tags from the given flat pattern. Returns NULL_TAG if the flat pattern does not contain bend lines.

Environment

Internal and External

History

Original release was in V13.0.

Required License(s)

gateway

```
int UF_SMD_ask_bend_lines
(
    tag_t flat_pattern,
    int * num_bend_lines,
    tag_p_t * bend_lines
)
```

tag_t	**flat_pattern**	Input	Object ID of the flat pattern.
int *	**num_bend_lines**	Output	Number of bend line object IDs.
tag_p_t *	**bend_lines**	Output to UF_*free*	Array of bend line object IDs. This array must be freed by calling UF_free.

UF_SMD_ask_bend_tan_lines (view source)

Defined in: uf_smd.h

Overview

Acquires an array of bend tangent line object tags from the given flat pattern. Returns NULL_TAG if the flat pattern does not contain bend tangent lines.

Environment

Internal and External

History

Original release was in V13.0.

Required License(s)

gateway

```
int UF_SMD_ask_bend_tan_lines
(
    tag_t flat_pattern,
    int * num_bend_tan_lines,
    tag_p_t * bend_tan_lines
)
```

tag_t	**flat_pattern**	Input	Object ID of the flat pattern.
int *	**num_bend_tan_lines**	Output	Number of bend tangent line object IDs.
tag_p_t *	**bend_tan_lines**	Output to UF_*free*	Array of bend tangent line object IDs. This array must be freed by calling UF_free.

UF_SMD_ask_child_pip_nodes (view source)

Defined in: uf_smd.h

Overview

This routine returns the array of PIP nodes that are children of the given parent node. If you want all processes, pass in NULL_TAG as the parent. Pass in a process tag to get an array of groups. Pass in a group tag to get an array of steps.

Environment

Internal and External

History

Original release was in NX 1.0.2.

Required License(s)

gateway

```
int UF_SMD_ask_child_pip_nodes
(
    tag_t parent,
    int * n_pip_nodes,
    tag_t * * pip_nodes
)
```

tag_t	**parent**	Input	The parent PIP node
int *	**n_pip_nodes**	Output	The size of the array of node tags
tag_t * *	**pip_nodes**	Output to UF_*free*	The array of child PIP nodes

UF_SMD_ask_contour_lines (view source)

Defined in: uf_smd.h

Overview

Acquires an array of contour line object tags from the given flat pattern. Returns NULL_TAG if the flat pattern does not contain contour lines.

Environment

Internal and External

History

Original release was in V13.0.

Required License(s)

gateway

```
int UF_SMD_ask_contour_lines
(
    tag_t flat_pattern,
    int * num_contour_lines,
    tag_p_t * contour_lines
)
```

tag_t	**flat_pattern**	Input	Object ID of the flat pattern.
int *	**num_contour_lines**	Output	Number of contour line object IDs.
tag_p_t *	**contour_lines**	Output to UF_*free*	Array of contour line object IDs. This array must be freed by calling UF_free.

UF_SMD_ask_flat_pattern (view source)

Defined in: uf_smd.h

Overview

Acquires the tag of current flat pattern group entity given the object identifier of a solid body. Returns NULL_TAG if a flat pattern does not exist for the associated body.

Environment

Internal and External

History

Original release was in V13.0.

Required License(s)

gateway

```
int UF_SMD_ask_flat_pattern
(
    tag_t body,
    tag_p_t flat_pattern
)
```

tag_t	**body**	Input	Object ID of a solid body.
tag_p_t	**flat_pattern**	Output	Object ID of the flat pattern group.

UF_SMD_ask_formable_feats (view source)

Defined in: uf_smd.h

Overview

Returns a linked list of all formable features used in the composition of the specified solid body. An example of a formable feature would be a flange. The handling of the linked list can be done by using the UF_MODL__list_ routines.

Environment

Internal and External

Required License(s)

gateway

```
int UF_SMD_ask_formable_feats
(
    tag_t body,
    uf_list_p_t * feat_list
)
```

tag_t	**body**	Input	Object ID of a solid body.
uf_list_p_t *	**feat_list**	Output to UF_*free*	A linked list of formable features which take part in the composition of the specified solid body. Use UF_MODL_delete_list to free the space allocated for the linked list.

UF_SMD_ask_formblk_lines (view source)

Defined in: uf_smd.h

Overview

Acquires an array of form block line object tags from the given flat pattern. Returns NULL_TAG if the flat pattern does not contain form block lines.

Environment

Internal and External

History

Original release was in V13.0.

Required License(s)

gateway

```
int UF_SMD_ask_formblk_lines
(
    tag_t flat_pattern,
    int * num_formblk_lines,
    tag_p_t * formblk_lines
)
```

tag_t	**flat_pattern**	Input	Object ID of the flat pattern.
int *	**num_formblk_lines**	Output	Number of form block line object IDs.
tag_p_t *	**formblk_lines**	Output to UF_*free*	Array of form block line object IDs. This array must be freed by calling UF_free.

UF_SMD_ask_fp_add_curves (view source)

Defined in: uf_smd.h

Overview

Acquires an array of object tags of additional curves which were mapped to the flat pattern associated with the given body. Returns NULL_TAG if the flat pattern does not contain additional curves.

Environment

Internal and External

History

Original release was in V13.0.

Required License(s)

gateway

```
int UF_SMD_ask_fp_add_curves
(
    tag_t body,
    int * num_additional_curves,
    tag_p_t * additional_curves
)
```

tag_t	**body**	Input	Object ID of the solid body.
int *	**num_additional_curves**	Output	Number of additional curve object IDs.
tag_p_t *	**additional_curves**	Output to UF_*free*	Array of additional curve object IDs. This array must be freed by calling UF_free.

UF_SMD_ask_fp_children (view source)

Defined in: uf_smd.h

Overview

Return the children of an object mapped by a flat pattern. The flat pattern specified must be the Master Flat Pattern which is the newest flat pattern for a given body. (This can be found using UF_SMD_ask_flat_pattern().) The parent specified may be an edge of the body being flattened or an additional curve (including Contour, Mold and Form Block curves). The returned children will be those members of the flat pattern group which correspond to the input object.

Returns

SMDFLAT_ERR_NOT_MASTER_FLAT_PATTERN

Environment

Internal and External

History

Original release was in V19.0.

Required License(s)

gateway

```
int UF_SMD_ask_fp_children
(
    tag_t flat_pattern,
    tag_t parent,
    int * num_children,
    tag_t * * children
)
```

tag_t	**flat_pattern**	Input	Object ID of the flat pattern.
tag_t	**parent**	Input	Object ID of the flat pattern.
int *	**num_children**	Output	Number of child object IDs in the flat pattern. (Or 0 if there is an error.)
tag_t * *	**children**	Output to UF_*free*	Array of child object IDs. This array must be freed by calling UF_free. (Or NULL if there is an error.)

UF_SMD_ask_fp_faces (view source)

Defined in: uf_smd.h

Overview

Acquires an array of object tags of the faces which were used to generate the flat pattern.

Environment

Internal and External

History

Original release was in V13.0.

Required License(s)

gateway

```
int UF_SMD_ask_fp_faces
(
    tag_t body,
    int * num_faces,
    tag_p_t * faces
)
```

tag_t	**body**	Input	Object ID of the solid body.
int *	**num_faces**	Output	Number of flat pattern faces.
tag_p_t *	**faces**	Output to UF_*free*	Array of flat pattern faces. This array must be freed by calling UF_free.

UF_SMD_ask_fp_parent (view source)

Defined in: uf_smd.h

Overview

Return the parent (edge or additional curve) of a curve or point in a flat pattern group. The object must be in the Master Flat Pattern which is the newest flat pattern for a given body. The returned parent is the object that corresponds with the child object. If the object is an additional curve (including Contour, Mold and Form Block curves) or point of the flat pattern the parent will be a curve or point, otherwise the parent will be an edge.

Returns

SMDFLAT_ERR_NOT_FLAT_PATTERN
SMDFLAT_ERR_NOT_MASTER_FLAT_PATTERN
SMDFLAT_ERR_NOT_MASTER_FLAT_PATTERN_ELEMENT

Environment

Internal and External

History

Original release was in V19.0.

Required License(s)

gateway

```
int UF_SMD_ask_fp_parent
(
    tag_t child,
    tag_p_t parent
)
```

tag_t	**child**	Input	Object ID of the child entity in the flat pattern.
tag_p_t	**parent**	Output	Parent entity of the input child. (Or NULL_TAG if there is an error.)

UF_SMD_ask_fp_prefs (view source)

Defined in: uf_smd.h

Overview

Inquires the flat pattern preferences currently available for Sheet Metal Design. The routine returns a data structure and individual elements that must be freed by a call to UF_free.

Environment

Internal and External

History

Original release was in V13.0.

Required License(s)

gateway

```
int UF_SMD_ask_fp_prefs
(
    UF_SMD_fp_prefs_p_t * fp_options
)
```

UF_SMD_fp_prefs_p_t *	**fp_options**	Output to UF_*free*	Structure that contains the SMD flat pattern preferences.

UF_SMD_ask_general_prefs (view source)

Defined in: uf_smd.h

Overview

Inquires the current settings of general SMD preferences. These preferences currently consist of the default bend allowance formula for the creation of a flat pattern grouped object, the specified material, and whether or not to display the bend sequence labels.

Environment

Internal and External

Required License(s)

gateway

```
int UF_SMD_ask_general_prefs
(
    UF_SMD_gen_prefs_p_t * general_prefs
)
```

UF_SMD_gen_prefs_p_t *	**general_prefs**	Output to UF_*free*	The address of a pointer to typed data.

UF_SMD_ask_pip_node_data (view source)

Defined in: uf_smd.h

Overview

Ask for data associated with a node on the Part In Process Navigator tree.

Environment

Internal and External

History

Original release was in NX 1.0.2.

Required License(s)

gateway

```
int UF_SMD_ask_pip_node_data
(
    tag_t node,
    UF_SMD_pip_node_data_t * node_data
)
```

tag_t	**node**	Input	The tag of the PIP node for which data is needed.
UF_SMD_pip_node_data_t *	**node_data**	Output	Data associated with this node.

UF_SMD_ask_standards (view source)

Defined in: uf_smd.h

Overview

Queries the standards file with the given query text string. This routine returns a data structure and individual elements that must be freed by a call to UF_SMD_free_std_results.

Environment

Internal and External

Required License(s)

gateway

```
int UF_SMD_ask_standards
(
    char * query_string,
    UF_SMD_results_p_t* results
)
```

char *	**query_string**	Input	Query string
UF_SMD_results_p_t*	**results**	Output to UF_*free*	Pointer to pointer to an array of results data structures. This must be freed by calling UF_SMD_free_std_results.

UF_SMD_chk_entire_part_stds (view source)

Defined in: uf_smd.h

Overview

Checks the entire part against the sheet metal standards file. This routine returns a data structure and individual elements that must be freed by a call to UF_SMD_free_std_chk_results.

Environment

Internal and External

Required License(s)

gateway

```
int UF_SMD_chk_entire_part_stds
(
    UF_SMD_chk_std_results_p_t * results
)
```

UF_SMD_chk_std_results_p_t *	**results**	Output to UF_*free*	Pointer to pointer to results data structure. This must be freed by calling UF_SMD_free_std_chk_results.

UF_SMD_chk_feature_stds (view source)

Defined in: uf_smd.h

Overview

This routine will utilize the "chk_std" function in the stds file to compare the feature's parameter values with the stds and return the test string from the standards file.

Environment

Internal and External

Required License(s)

gateway

```
int UF_SMD_chk_feature_stds
(
    tag_t * features,
    int num_features,
    UF_SMD_chk_std_results_p_t * results
)
```

tag_t *	**features**	Input	Array of feature tags
int	**num_features**	Input	Number of features
UF_SMD_chk_std_results_p_t *	**results**	Output to UF_*free*	Ptr to results. This must be freed by calling UF_SMD_free_std_chk_results

UF_SMD_create_flat_pattern (view source)

Defined in: uf_smd.h

Overview

Creates a flat pattern grouped object (collection of wireframe entities) for the solid body associated with the root face. A NULL_TAG is returned if a flat pattern is not generated from the solid body. If a non-fatal error occurs during the creation of a flat pattern grouped object, a valid tag is returned along with an error code.

Environment

Internal and External

History

Original release was in V13.0.

Required License(s)

sheet_metal_design

```
int UF_SMD_create_flat_pattern
(
    char * name,
    int layer,
    tag_t start_face,
    tag_p_t flat_pattern
)
```

char *	**name**	Input	Name of the flat pattern
int	**layer**	Input	Layer on which to create the flat pattern grouped object.
tag_t	**start_face**	Input	Object ID of the start face for traversing the part to create the flat pattern.
tag_p_t	**flat_pattern**	Output	Object ID of the flat pattern grouped object.

UF_SMD_create_pip_node (view source)

Defined in: uf_smd.h

Overview

Create a node (process, group, or step) on the Part In Process Navigator tree.

Environment

Internal and External

History

Original release was in NX 1.0.2.

Required License(s)

sheet_metal_design

```
int UF_SMD_create_pip_node
(
    UF_SMD_pip_node_data_p_t node_data,
    tag_t * node_tag
)
```

UF_SMD_pip_node_data_p_t	**node_data**	Input	Data used to create node The parent of a UF_SMD_pip_process must be NULL_TAG. The parent of a UF_SMD_pip_group must be a process. The parent of a UF_SMD_pip_step must be a group. UF_SMD_pip_unform_process and UF_SMD_pip_form_process node are created automatically and cannot be created with this function.
tag_t *	**node_tag**	Output	Tag of the new PIP node

UF_SMD_create_representation (view source)

Defined in: uf_smd.h

Overview

This routine creates a Sheet Metal Representation of a Modeling feature.
It currently supports Modeling Hole and Slot features.

Environment

Internal and External

Required License(s)

sheet_metal_design

```
int UF_SMD_create_representation
(
    tag_t feature,
    int hint_flag,
    tag_t * new_feature
)
```

tag_t	**feature**	Input	the tag of Modeling feature
int	**hint_flag**	Input	hint flag CURRENTLY UNUSED
tag_t *	**new_feature**	Output	the tag of output SMD feature CURRENTLY EQUALS MODELING FEATURE TAG

UF_SMD_delete_flat_pattern (view source)

Defined in: uf_smd.h

Overview

Deletes the flat pattern grouped object along with the associated wireframe geometric entities.

Environment

Internal and External

History

Original release was in V13.0.

Required License(s)

sheet_metal_design

```
int UF_SMD_delete_flat_pattern
(
    tag_t flat_pattern
)
```

tag_t	**flat_pattern**	Input	Body of the flat pattern to be deleted.

UF_SMD_delete_pip_node (view source)

Defined in: uf_smd.h

Overview

Delete the specified PIP node. When you delete a Process, all children Groups and Steps are deleted. When you delete a Group, its Steps are deleted.

Environment

Internal and External

History

Original release was in NX 1.0.2.

Required License(s)

sheet_metal_design

```
int UF_SMD_delete_pip_node
(
    tag_t node
)
```

tag_t	**node**	Input	The tag of the PIP node to be deleted.

UF_SMD_delete_representation (view source)

Defined in: uf_smd.h

Overview

This routine deletes a Sheet Metal Representation of a Modeling feature.
It currently supports SMD Hole and Slot representations.

Environment

Internal and External

Required License(s)

sheet_metal_design

```
int UF_SMD_delete_representation
(
    tag_t feature,
    tag_t * new_feature
)
```

tag_t	**feature**	Input	the tag of SMD feature
tag_t *	**new_feature**	Output	the tag of the output Modeling feature CURRENTLY EQUALS SMD FEATURE TAG

UF_SMD_edit_pip_node (view source)

Defined in: uf_smd.h

Overview

This routine edits the data associated with a PIP node.

Environment

Internal and External

History

Original release was in NX 1.0.2.

Required License(s)

sheet_metal_design

```
int UF_SMD_edit_pip_node
(
    tag_t node,
    UF_SMD_pip_node_data_t * node_data
)
```

tag_t	**node**	Input	The tag of the PIP node for which data is to be edited.
UF_SMD_pip_node_data_t *	**node_data**	Input	New data associated with this node. Note that you cannot change the node_type or the parent. These fields will be ignored.

UF_SMD_eval_trim_angles (view source)

Defined in: uf_smd.h

Overview

This routine will return the total number of points created. Associated to each point is a point tag, curve tag, face tag and calculated trim angle.

Environment

Internal and External

Required License(s)

sheet_metal_design

```
int UF_SMD_eval_trim_angles
(
    tag_p_t curves,
    int num_curves,
    tag_p_t faces,
    int num_faces,
    double dir_vec [ 3 ] ,
    double equal_arc_len,
    int * num_of_pts,
    tag_t * * pts,
    tag_t * * assoc_curves,
    tag_t * * assoc_faces,
    double * * trim_ang_array
)
```

tag_p_t	**curves**	Input	array of curves
int	**num_curves**	Input	number of curves/edges
tag_p_t	**faces**	Input	array of faces
int	**num_faces**	Input	number of faces
double	**dir_vec [3]**	Input	trimming direction

double	**equal_arc_len**	Input	equal arc length
int *	**num_of_pts**	Output	number of points
tag_t * *	**pts**	Output to UF_*free*	Array of point tags. This array must be freed by calling UF_free.
tag_t * *	**assoc_curves**	Output to UF_*free*	Array of associated curves tags. This array must be freed by calling UF_free.
tag_t * *	**assoc_faces**	Output to UF_*free*	Array of associated faces tags . This array must be freed by calling UF_free.
double * *	**trim_ang_array**	Output to UF_*free*	Array of trim angles at each point. This array must be freed by calling UF_free.

UF_SMD_exec_formed (view source)

Defined in: uf_smd.h

Overview

Execute to the FORMED sequence in the forming table. This results in the body being placed in its fully formed state (folded representation).

Environment

Internal and External

Required License(s)

sheet_metal_design

```
int UF_SMD_exec_formed
(
    tag_t body
)
```

tag_t	**body**	Input	Object ID of current form table.

UF_SMD_exec_unformed (view source)

Defined in: uf_smd.h

Overview

Execute to the UNFORMED sequence in the forming table. This results in the body being placed in its fully unformed state (flattened representation).

Environment

Internal and External

Required License(s)

sheet_metal_design

```
int UF_SMD_exec_unformed
(
    tag_t body
)
```

tag_t	**body**	Input	Object ID of current form table.

UF_SMD_execute_pip_node (view source)

Defined in: uf_smd.h

Overview

Execute the specified PIP node. Note that PIP Processes cannot be executed. If you execute a Group, all Steps within the Group are executed.

Environment

Internal and External

History

Original release was in NX 1.0.2.

Required License(s)

sheet_metal_design

```
int UF_SMD_execute_pip_node
(
    tag_t node
)
```

tag_t	**node**	Input	The tag of the PIP node for which data is needed.

UF_SMD_free_std_chk_results (view source)

Defined in: uf_smd.h

Overview

Frees all memory associated with the results of a standards check.

Environment

Internal and External

Required License(s)

sheet_metal_design

```
int UF_SMD_free_std_chk_results
(
    UF_SMD_chk_std_results_t * results
)
```

UF_SMD_chk_std_results_t *	**results**	Input	Pointer to check standard results structure

UF_SMD_free_std_results (view source)

Defined in: uf_smd.h

Overview

Frees all memory associated with the results of a standards query.

Environment

Internal and External

Required License(s)

sheet_metal_design

```
int UF_SMD_free_std_results
(
    UF_SMD_results_t * results
)
```

UF_SMD_results_t *	**results**	Input	Pointer to results structure

UF_SMD_initialize (view source)

Defined in: uf_smd.h

Overview

Initializes the SMD environment and verifies that an SMD license exists. If you do not call UF_SMD_initialize, then subsequent calls to SMD Open API routines fail with an error code of UF_err_program_not_initialized.

Environment

Internal and External

Required License(s)

sheet_metal_design

```
int UF_SMD_initialize
(
    void
)
```

UF_SMD_is_initialized (view source)

Defined in: uf_smd.h

Overview

Checks if the SMD environment has been successfully initialized.

Return

Return Code
0 = SMD is not initialized
1 = SMD has been properly initialized

Environment

Internal and External

Required License(s)

gateway

```
int UF_SMD_is_initialized
(
    void
)
```

UF_SMD_load_standards (view source)

Defined in: uf_smd.h

Overview

Load the given standards file into memory. If no file specification is given, then it loads the file pointed to by $UGII_UGSMD_STDS_LIBRARY. If the environment variable doesn't point to a specific file, the routine will load the file ugsmd_def.std from one of the following directories, searching them in order:
current working directory
$UGII_UGSMD_STDS_LIBRARY
$UGII_BASE_DIR/ugii

Environment

Internal and External

Required License(s)

gateway

```
int UF_SMD_load_standards
(
    char * fspec
)
```

char *	**fspec**	Input	File specification of standards file, or "" to use default standards file.

UF_SMD_set_fp_prefs (view source)

Defined in: uf_smd.h

Overview

Defines the flat pattern preferences currently available for Sheet Metal Design. The routine requires that the data structure and individual elements be allocated by the calling routine.

Environment

Internal and External

History

Original release was in V13.0.

Required License(s)

sheet_metal_design

```
int UF_SMD_set_fp_prefs
(
    UF_SMD_fp_prefs_p_t fp_options
)
```

UF_SMD_fp_prefs_p_t	**fp_options**	Input	Structure that contains the SMD flat pattern preferences.

UF_SMD_set_general_prefs (view source)

Defined in: uf_smd.h

Overview

Defines the general SMD preferences. These preferences currently consist of the default bend allowance formula for the creation of a flat pattern grouped object, the specified material, and whether or not to display the bend sequence labels.

(Radius + (Thickness 0.44))rad(Angle)

The default bend formula must be a valid RHS (right hand side) for an NX Expression.

Radius, Thickness, and Angle when used, must be represented exactly as shown with the first letter capitalized. They are not actually expression variables in the usual sense, but are substituted variables for each flange when the expression is evaluated.

Radius is the inside bend radius; Thickness is the material thickness; Angle is the bend angle; and rad is a built-in NX Expression function. The value .44 represents the Neutral Axis Factor, an empirical value which may be adjusted for the material type and thickness to correctly calculate the developed length of the bend portion of a flange feature due to folding.

Environment

Internal and External

Required License(s)

sheet_metal_design

```
int UF_SMD_set_general_prefs
(
    UF_SMD_gen_prefs_p_t general_prefs
)
```

UF_SMD_gen_prefs_p_t	**general_prefs**	Input	A pointer to typed data.

UF_SMD_terminate (view source)

Defined in: uf_smd.h

Overview

Terminates the SMD environment. Once UF_SMD_terminate is called from an external Open API program, the program should exit. It is recommended that you do not attempt to call UF_SMD_initialize again and continue operation.

Environment

Internal and External

Required License(s)

sheet_metal_design

```
int UF_SMD_terminate
(
    void
)
```

UF_SMD_update_flat_pattern (view source)

Defined in: uf_smd.h

Overview

Determines if any modification to the solid body has been performed and then updates the flat pattern accordingly. A pointer to a NULL_TAG is returned if a flat pattern is not generated from the solid body.

Environment

Internal and External

History

Original release was in V13.0.

Required License(s)

sheet_metal_design

```
int UF_SMD_update_flat_pattern
(
    tag_t body,
    tag_p_t flat_pattern
)
```

tag_t	**body**	Input	Object ID of the body that the flat pattern was created from.
tag_p_t	**flat_pattern**	Output	Object ID of the flat pattern group object.