UF_PARAM_append_ude (view source)

Defined in: uf_param.h

Overview
This function appends the User Defined Machine Contol Event with name 'ude_name' to the Machine Control set of type 'ude_set_type' in the param object 'param'. It also returns the created User Defined Machine Control Event object in 'ude_obj'.

Environment
Internal and External

History
Originally released in V18.0

```
int UF_PARAM_append_ude
(
    tag_t param,
    UF_UDE_set_type_t ude_set_type,
    char * ude_name,
    UF_UDE_t * ud_obj
)
```

tag_t	**param**	Input	see above
UF_UDE_set_type_t	**ude_set_type**	Input	see above
char *	**ude_name**	Input	see above
UF_UDE_t *	**ud_obj**	Output	see above

UF_PARAM_ask_2d_value (view source)

Defined in: uf_param.h

Overview
This function returns in 'value' the 2d array of the parameter specified by 'param_index'. It is the value of this parameter that is currently being used by the object specified by 'param_tag'.

Environment
Internal and External

History
Originally released in NX2.0

```
int UF_PARAM_ask_2d_value
(
    tag_t param_tag,
    int param_index,
    double value [ 2 ]
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
double	**value [2]**	Output	- see above

UF_PARAM_ask_3d_value (view source)

Defined in: uf_param.h

Overview
This function returns in 'value' the 3d array of the parameter specified by 'param_index'. It is the value of this parameter that is currently being used by the object specified by 'param_tag'.

Environment
Internal and External

History
Originally released in NX2.0

```
int UF_PARAM_ask_3d_value
(
    tag_t param_tag,
    int param_index,
    double value [ 3 ]
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
double	**value [3]**	Output	- see above

UF_PARAM_ask_double_value (view source)

Defined in: uf_param.h

Overview
This function returns in 'value' the value of the parameter specified by 'param_index'. It is the value of this parameter that is currently being used by the object specified by 'param_tag'.

Environment
Internal and External

History
Originally released in V16.0

```
int UF_PARAM_ask_double_value
(
    tag_t param_tag,
    int param_index,
    double * value
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
double *	**value**	Output	- see above

UF_PARAM_ask_double_vla (view source)

Defined in: uf_param.h

Overview

This function returns in 'dbl_array' the array of doubles of the parameter specified by 'param_index'. It is the array of values of this parameter that is currently being used by the object specified by 'param_tag'. The number of doubles is returned in 'count.'

Environment

Internal and External

History

Originally released in NX3.0

```
int UF_PARAM_ask_double_vla
(
    tag_t param_tag,
    int param_index,
    int * count,
    double * * dbl_array
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
int *	**count**	Output	- number of doubles in dbl_array
double * *	**dbl_array**	Output to UF_*free*	- see above. This must be freed by the caller using UF_free

---

UF_PARAM_ask_inherited_params (view source)

Defined in: uf_param.h

Overview

This function returns back the list of inherited paramter indices of the object specifed in 'param_tag'. This is parameters that are currently inherited. Since data inheritance is dynamic this list can change each time this function is called. The number of indices is returned in 'count' and the list of indices is returned in 'indices'.

Environment
Internal and External

History
Originally released in V16.0

```
int UF_PARAM_ask_inherited_params
(
    tag_t param_tag,
    int * count,
    int indices [ UF_PARAM_REQUIRED_LIST_SIZE ]
)
```

tag_t	**param_tag**	Input	- see above
int *	**count**	Output	- see above
int	**indices [UF_PARAM_REQUIRED_LIST_SIZE]**	Output	- see above

---

UF_PARAM_ask_int_value (view source)

Defined in: uf_param.h

Overview
This function returns in 'value' the value of the parameter specified by 'param_index'. It is the value of this parameter that is currently being used by the object specified by 'param_tag'.

Environment
Internal and External

History
Originally released in V16.0

```
int UF_PARAM_ask_int_value
(
    tag_t param_tag,
    int param_index,
    int * value
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
int *	**value**	Output	- see above

UF_PARAM_ask_int_vla (view source)

Defined in: uf_param.h

Overview

This function returns in 'int_array' the array of integer values of the parameter specified by 'param_index'. It is the array of values of this parameter that is currently being used by the object specified by 'param_tag'. The number of integer values is returned in 'count.'

Environment

Internal and External

History

Originally released in NX3.0

```
int UF_PARAM_ask_int_vla
(
    tag_t param_tag,
    int param_index,
    int * count,
    int * * int_array
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
int *	**count**	Output	- number of integers in int_array
int * *	**int_array**	Output to UF_*free*	- see above. This must be freed by the caller using UF_free

UF_PARAM_ask_logical_value (view source)

Defined in: uf_param.h

Overview

This function returns in 'value' the value of the parameter specified by 'param_index'. It is the value of this parameter that is currently being used by the object specified by 'param_tag'.

Environment

Internal and External

History

Originally released in V16.0

```
int UF_PARAM_ask_logical_value
(
```

```
    tag_t param_tag,
    int param_index,
    logical * value
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
logical *	**value**	Output	- see above

UF_PARAM_ask_param_attributes (view source)

Defined in: uf_param.h

Overview

This function returns in 'attributes' the data documented in the structure UF_PARAM_index_attribute_t. This data will be for the parameter specified by 'param_index'.

Environment

Internal and External

History

Originally released in V16.0

```
int UF_PARAM_ask_param_attributes
(
    int param_index,
    UF_PARAM_index_attribute_t * attributes
)
```

int	**param_index**	Input	- see above
UF_PARAM_index_attribute_t *	**attributes**	Output	- see above

---

UF_PARAM_ask_param_definer (view source)

Defined in: uf_param.h

Overview

This function returns in 'definer_tag' the object from which object 'param_tag' is currently getting the value of the parameter 'param_index'. It is possible that 'definer_tag' is the same as 'param_tag'.

Environment

Internal and External

History

Originally released in V16.0

```
int UF_PARAM_ask_param_definer
(
    tag_t param_tag,
    int param_index,
    tag_t * definer_tag
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
tag_t *	**definer_tag**	Output	- see above

UF_PARAM_ask_param_status (view source)

Defined in: uf_param.h

Overview

This function returns in 'status' the current inheritance status of the parameter 'param_index' for the object 'param_tag'.

Environment

Internal and External

History

Originally released in V16.0

```
int UF_PARAM_ask_param_status
(
    tag_t param_tag,
    int param_index,
    UF_PARAM_status_t * status
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
UF_PARAM_status_t *	**status**	Output	- see above

UF_PARAM_ask_required_params (view source)

Defined in: uf_param.h

Overview

This function returns back the list of required paramter indices for the object specifed in 'param_tag'. The number of indices is returned in 'count'

and the list of indices is returned in 'indices'.

Environment
Internal and External

History
Originally released in V16.0

int UF_PARAM_ask_required_params
(
tag_t param_tag,
int * count,
int * * indices
)

tag_t	**param_tag**	Input	- see above
int *	**count**	Output	- see above
int * *	**indices**	Output to UF_*free*	- see above

UF_PARAM_ask_str_value (view source)

Defined in: uf_param.h

Overview
This function returns in 'value' the value of the parameter specified by 'param_index'. It is the value of this parameter that is currently being used by the object specified by 'param_tag'.

Environment
Internal and External

History
Originally released in V16.0

int UF_PARAM_ask_str_value
(
tag_t param_tag,
int param_index,
char value [MAX_LINE_BUFSIZE]
)

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
char	**value [MAX_LINE_BUFSIZE]**	Output	- see above

UF_PARAM_ask_subobj_ptr_value (view source)

Defined in: uf_param.h

Overview

This function copies to 'value' the contents of the memory pointed at by the pointer parameter specified by 'param_index'. It is the value of this parameter that is currently being used by the object specified by 'param_tag'.

The caller must ensure that value points at a UF_PARAM_<type>_t that corresponds to the type of object that is pointed at by this parameter index. The proper type of object is specified in uf_param_indices.h in the comments about the parameter index.

Environment

Internal and External

History

Originally released in V16.0

```
int UF_PARAM_ask_subobj_ptr_value
(
    tag_t param_tag,
    int param_index,
    void * value
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
void *	**value**	Output	- see above

UF_PARAM_ask_tag_value (view source)

Defined in: uf_param.h

Overview

This function returns in 'value' the value of the parameter specified by 'param_index'. It is the value of this parameter that is currently being used by the object specified by 'param_tag'.

Environment

Internal and External

History

Originally released in V16.0

```
int UF_PARAM_ask_tag_value
(
    tag_t param_tag,
    int param_index,
    tag_t * value
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
tag_t *	**value**	Output	- see above

UF_PARAM_ask_tag_vla (view source)

Defined in: uf_param.h

Overview

This function returns in 'tag_array' the array of tags of the parameter specified by 'param_index'. It is the array of values of this parameter that is currently being used by the object specified by 'param_tag'. The number of tags is returned in 'count.'

Environment

Internal and External

History

Originally released in NX3.0

```
int UF_PARAM_ask_tag_vla
(
    tag_t param_tag,
    int param_index,
    int * count,
    tag_t * * tag_array
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
int *	**count**	Output	- number of tags in tag_array
tag_t * *	**tag_array**	Output to UF_*free*	- see above. This must be freed by the caller using UF_free

---

UF_PARAM_ask_udes (view source)

Defined in: uf_param.h

Overview

This function returns all the User Defined Machine Control Event objects in 'ude_objs' contained in the Machine Control Event set of type 'ude_set_type' in the param object 'param'. It also returns the number of User Defined Machine Control Events in 'num_of_udes'.

Environment

Internal and External

See Also
Functions in the UF_UDE module to interrogate or edit any of the 'ude_objs'

History
Originally released in V18.0

int UF_PARAM_ask_udes
(
tag_t param,
UF_UDE_set_type_t ude_set_type,
int * num_of_udes,
UF_UDE_t * * ude_objs
)

tag_t	**param**	Input	see above
UF_UDE_set_type_t	**ude_set_type**	Input	see above
int *	**num_of_udes**	Output	see above
UF_UDE_t * *	**ude_objs**	Output to UF_*free*	see above This should be freed by calling UF_free.

UF_PARAM_can_accept_ude (view source)

Defined in: uf_param.h

Overview
This function sets the 'response' to TRUE if the specified User Defined Machine Control of name 'ude_name' is valid for the Machine Control set of type 'set_type' in the param object 'param' or to FALSE if it is not valid.

Environment
Internal and External

History
Originally released in V18.0

int UF_PARAM_can_accept_ude
(
tag_t param,
UF_UDE_set_type_t ude_set_type,
char * ude_name,
logical * response
)

tag_t	**param**	Input	- see above
UF_UDE_set_type_t	**ude_set_type**	Input	- see above

char *	**ude_name**	Input	- see above
logical *	**response**	Output	- see above

UF_PARAM_can_accept_ude_set (view source)

Defined in: uf_param.h

Overview
This function sets the 'response' to TRUE if the Machine Control 'set_type' is valid for the param object 'param' or to FALSE if it is not valid.

Environment
Internal and External

History
Originally released in V18.0

```
int UF_PARAM_can_accept_ude_set
(
    tag_t param,
    UF_UDE_set_type_t ude_set_type,
    logical * response
)
```

tag_t	**param**	Input	- see above
UF_UDE_set_type_t	**ude_set_type**	Input	- see above
logical *	**response**	Output	- see above

UF_PARAM_check (view source)

Defined in: uf_param.h

Overview
This function will check if all the parameters of UF_PARAM object 'param' are in a state which is valid for generation. If they are then TRUE is returned in 'is_ok' else FALSE is returned in 'is_ok'. What 'generation' means is determined by the type of the UF_PARAM object. E.g., if it is a UF_OPER object then 'generation' means tool path generation.

Environment
Internal and External

History
Originally released in V16.0

```
int UF_PARAM_check
(
    tag_t param,
    logical * is_ok
)
```

tag_t	**param**	Input	- see above
logical *	**is_ok**	Output	- see above

UF_PARAM_delete_all_udes (view source)

Defined in: uf_param.h

Overview
This deletes all the User Defined Machine Control Event objects contained in the Machine Control Event set of type 'ude_set_type' in the param object 'param'.

Environment
Internal and External

History
Originally released in V18.0

```
int UF_PARAM_delete_all_udes
(
    tag_t param,
    UF_UDE_set_type_t ude_set_type
)
```

tag_t	**param**	Input	see above
UF_UDE_set_type_t	**ude_set_type**	Input	see above

UF_PARAM_delete_ude (view source)

Defined in: uf_param.h

Overview
This function deletes the specified User Defined Machine Control Event object 'ude_obj' from the Machine Control Event set of type 'ude_set_type' in the param object 'param'.

Environment
Internal and External

History
Originally released in V18.0

```
int UF_PARAM_delete_ude
(
    tag_t param,
    UF_UDE_set_type_t ude_set_type,
    UF_UDE_t ude_obj
)
```

tag_t	**param**	Input	see above
UF_UDE_set_type_t	**ude_set_type**	Input	see above
UF_UDE_t	**ude_obj**	Input	see above

UF_PARAM_duplicate (view source)

Defined in: uf_param.h

Overview

This function creates a new object of the same type as 'old_obj_tag'. It initializes this object with the data of 'old_obj_tag'. It assigns 'name' to the new object as it's name. It returns the tag of this newly created object in 'new_obj_tag'.

Environment

Internal and External

History

Originally released in V16.0

```
int UF_PARAM_duplicate
(
    tag_t old_obj_tag,
    const char * name,
    tag_t * new_obj_tag
)
```

tag_t	**old_obj_tag**	Input	- see above
const char *	**name**	Input	- see above
tag_t *	**new_obj_tag**	Output	- see above

UF_PARAM_generate (view source)

Defined in: uf_param.h

Overview

This function generates the tool path for the operations associated with the object specified by 'param_tag'. If this object is a group then the tool paths for all operations in the group will be generated. If the object is an

operation then the tool path for that single operation will be generated.

Environment
Internal and External

History
Originally released in V16.0

int UF_PARAM_generate
(
tag_t param_tag,
logical * generated
)

tag_t	**param_tag**	Input	- see above
logical *	**generated**	Output	- see above

UF_PARAM_inherit_value (view source)

Defined in: uf_param.h

Overview
This function deletes any override that object 'param_tag' might have for parameter 'param_index'. Whether or not 'param_tag' was overriding the value of this parameter, it will be inheriting its value after this call.

Environment
Internal and External

History
Originally released in V16.0

int UF_PARAM_inherit_value
(
tag_t param_tag,
int param_index
)

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above

UF_PARAM_is_inherited (view source)

Defined in: uf_param.h

Overview
This function returns TRUE in 'answer' if the object specified by 'param_tag' is currently inheriting the parameter specified by 'param index'.

Environment
Internal and External

History
Originally released in V16.0

int UF_PARAM_is_inherited
(
tag_t param_tag,
int param_index,
logical * answer
)

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
logical *	**answer**	Output	- see above

UF_PARAM_is_load_with_parent (view source)

Defined in: uf_param.h

Overview
This function sets the 'response' to TRUE if the specified param object is set as load with parent or to FALSE otherwise.

Environment
Internal and External

History
Originally released in V18.0

int UF_PARAM_is_load_with_parent
(
tag_t param,
logical * response
)

tag_t	**param**	Input	- see above
logical *	**response**	Output	- see above

UF_PARAM_is_same_class (view source)

Defined in: uf_param.h

Overview

This function returns TRUE in 'answer' if the object specified by 'obj1_tag' is the same class as the object specified by 'obj2_tag'.

Environment
Internal and External

History
Originally released in V16.0

```
int UF_PARAM_is_same_class
(
    tag_t obj1_tag,
    tag_t obj2_tag,
    logical * answer
)
```

tag_t	**obj1_tag**	Input	- see above
tag_t	**obj2_tag**	Input	- see above
logical *	**answer**	Output	- see above

UF_PARAM_is_template (view source)

Defined in: uf_param.h

Overview
This function sets the 'response' to TRUE if the specified param object is set as template or to FALSE otherwise.

Environment
Internal and External

History
Originally released in V18.0

```
int UF_PARAM_is_template
(
    tag_t param,
    logical * response
)
```

tag_t	**param**	Input	- see above
logical *	**response**	Output	- see above

UF_PARAM_reinit (view source)

Defined in: uf_param.h

Overview
This function will reinitialize the data of 'param_to_reinit' with the data of 'param_to_init_from'.

Environment
Internal and External

History
Originally released in V16.0

```
int UF_PARAM_reinit
(
    tag_t param_to_reinit,
    tag_t param_to_reinit_from
)
```

tag_t	**param_to_reinit**	Input	- the param to reinit
tag_t	**param_to_reinit_from**	Input	- the param whose data is used to reinit param_to_reinit

---

UF_PARAM_rename (view source)

Defined in: uf_param.h

Overview
This function assigns the name 'new_name' to the object specified by 'param_tag'. The first character of 'new_name' must be an alphabetic character.

Environment
Internal and External

History
Originally released in V16.0

```
int UF_PARAM_rename
(
    tag_t param_tag,
    const char * new_name
)
```

tag_t	**param_tag**	Input	- see above
const char *	**new_name**	Input	- see above

UF_PARAM_replay_path (view source)

Defined in: uf_param.h

Overview

This function replays the tool path for the operations associated with the object specified by 'param_tag'. If this object is a group then the tool paths for all operations in the group will be replayed. If the object is an operation then the tool path for that single operation will be replayed.

Environment

Internal and External

History

Originally released in NX3

```
int UF_PARAM_replay_path
(
    tag_t param_tag
)
```

tag_t	**param_tag**	Input	- see above

UF_PARAM_set_2d_value (view source)

Defined in: uf_param.h

Overview

This function assigns the 2d array 'value' to the parameter specified by 'param_index' for the object specified by 'param_tag'. This has the effect of 'param_tag' overriding the inherited value of parameter 'param_index'.

Environment

Internal and External

History

Originally released in NX2.0

```
int UF_PARAM_set_2d_value
(
    tag_t param_tag,
    int param_index,
    double value [ 2 ]
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
double	**value [2]**	Input	- see above

UF_PARAM_set_3d_value (view source)

Defined in: uf_param.h

Overview
This function assigns the 3d array 'value' to the parameter specified by 'param_index' for the object specified by 'param_tag'. This has the effect of 'param_tag' overriding the inherited value of parameter 'param_index'.

Environment
Internal and External

History
Originally released in NX2.0

```
int UF_PARAM_set_3d_value
(
    tag_t param_tag,
    int param_index,
    double value [ 3 ]
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
double	**value [3]**	Input	- see above

UF_PARAM_set_double_value (view source)

Defined in: uf_param.h

Overview
This function assigns the value 'value' to the parameter specified by 'param_index' for the object specified by 'param_tag'. This has the effect of 'param_tag' overriding the inherited value of parameter 'param_index'.

Environment
Internal and External

History
Originally released in V16.0

```
int UF_PARAM_set_double_value
(
    tag_t param_tag,
    int param_index,
    double value
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above

double	**value**	Input	- see above

UF_PARAM_set_double_vla (view source)

Defined in: uf_param.h

Overview
This function assigns the array values defined in 'dbl_array' to the parameter specified by 'param_index' for the object specified by 'param_tag'. This has the effect of 'param_tag' overriding the inherited value of the parameter 'param_index.' The number of doubles to be assigned is in 'count.'

Environment
Internal and External

History
Originally released in NX3.0

```
int UF_PARAM_set_double_vla
(
    tag_t param_tag,
    int param_index,
    int count,
    double * dbl_array
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
int	**count**	Input	- number of doubles in dbl_array
double *	**dbl_array**	Input	- see above.

UF_PARAM_set_int_value (view source)

Defined in: uf_param.h

Overview
This function assigns the value 'value' to the parameter specified by 'param_index' for the object specified by 'param_tag'. This has the effect of 'param_tag' overriding the inherited value of parameter 'param_index'.

Environment
Internal and External

History
Originally released in V16.0

```
int UF_PARAM_set_int_value
(
    tag_t param_tag,
    int param_index,
    int value
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
int	**value**	Input	- see above

UF_PARAM_set_int_vla (view source)

Defined in: uf_param.h

Overview

This function assigns the array values defined in 'int_array' to the parameter specified by 'param_index' for the object specified by 'param_tag'. This has the effect of 'param_tag' overriding the inherited value of the parameter 'param_index.' The number of integers to be assigned is in 'count.'

Environment

Internal and External

History

Originally released in NX3.0

```
int UF_PARAM_set_int_vla
(
    tag_t param_tag,
    int param_index,
    int count,
    int * int_array
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
int	**count**	Input	- number of integers in int_array
int *	**int_array**	Input	- see above.

UF_PARAM_set_logical_value (view source)

Defined in: uf_param.h

Overview

This function assigns the value 'value' to the parameter specified by 'param_index' for the object specified by 'param_tag'. This has the effect of 'param_tag' overriding the inherited value of parameter 'param_index'.

Environment
Internal and External

History
Originally released in V16.0

```
int UF_PARAM_set_logical_value
(
    tag_t param_tag,
    int param_index,
    logical value
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
logical	**value**	Input	- see above

UF_PARAM_set_str_value (view source)

Defined in: uf_param.h

Overview
This function assigns the value 'value' to the parameter specified by 'param_index' for the object specified by 'param_tag'. This has the effect of 'param_tag' overriding the inherited value of parameter 'param_index'.

Environment
Internal and External

History
Originally released in V16.0

```
int UF_PARAM_set_str_value
(
    tag_t param_tag,
    int param_index,
    char * value
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
char *	**value**	Input	- see above

UF_PARAM_set_subobj_ptr_value (view source)

Defined in: uf_param.h

Overview

This function sets the value of the specified 'parm_index' to the contents of the memory pointed at by 'value'. This will be the value for 'parm_index' that will be used by the object specified by 'param_tag'.

The caller must ensure that value points at a UF_PARAM_<type>_t that corresponds to the type of object that is pointed at by this parameter index. The proper type of object is specified in uf_param_indices.h in the comments about the parameter index.

Environment

Internal and External

History

Originally released in V16.0

```
int UF_PARAM_set_subobj_ptr_value
(
    tag_t param_tag,
    int param_index,
    void * value
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
void *	**value**	Input	- see above

UF_PARAM_set_tag_value (view source)

Defined in: uf_param.h

Overview

This function assigns the value 'value' to the parameter specified by 'param_index' for the object specified by 'param_tag'. This has the effect of 'param_tag' overriding the inherited value of parameter 'param_index'.

Environment

Internal and External

History

Originally released in V16.0

```
int UF_PARAM_set_tag_value
(
    tag_t param_tag,
    int param_index,
    tag_t value
```

```
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
tag_t	**value**	Input	- see above

UF_PARAM_set_tag_vla (view source)

Defined in: uf_param.h

Overview

This function assigns the array values defined in 'tag_array' to the parameter specified by 'param_index' for the object specified by 'param_tag'. This has the effect of 'param_tag' overriding the inherited value of the parameter 'param_index.' The number of tags to be assigned is in 'count.'

Environment

Internal and External

History

Originally released in NX3.0

```
int UF_PARAM_set_tag_vla
(
    tag_t param_tag,
    int param_index,
    int count,
    tag_t * tag_array
)
```

tag_t	**param_tag**	Input	- see above
int	**param_index**	Input	- see above
int	**count**	Input	- number of tags in tag_array
tag_t *	**tag_array**	Input	- see above.
