

**/\* Task Operations \*/**

**#ifdef \_\_ANSI\_\_**

```
extern int oktcre( char *name, prio priority, int stacksize, bit_field
                  mode, bit_field options, task_id tid ) ;
extern int oktdel( task_id tid ) ;
extern int oktidt( char *name, node_id nid, task_id tid ) ;
extern int oktsta( task_id tid, void start(void *), void *arguments,
                  int arg_length ) ;
extern int oktrst( task_id tid, void *arguments, int arg_length ) ;
extern int oktsus( task_id tid ) ;
extern int oktrsm( task_id tid ) ;
extern int oktspr( task_id tid, prio new_prio, prio *old_prio ) ;
extern int oktsmd( bit_field new_mode, bit_field mask, bit_field
                  *old_mode ) ;
extern int oktrnp( task_id tid, int loc_number, word *loc_value ) ;
extern int oktwnp( task_id tid, int loc_number, word loc_value ) ;
extern int oktinf( task_id tid, prio *priority, bit_field *mode,
                  bit_field *options, bit_field *event, bit_field
                  *exception, int state );
```

**#else**

```
extern int oktcre( ) ;
extern int oktdel( ) ;
extern int oktidt( ) ;
extern int oktsta( ) ;
extern int oktrst( ) ;
extern int oktsus( ) ;
extern int oktrsm( ) ;
extern int oktspr( ) ;
extern int oktsmd( ) ;
extern int oktrnp( ) ;
extern int oktwnp( ) ;
extern int oktinf( ) ;
```

**#endif**

```
#define task_create      oktcre
#define task_delete     oktdel
#define task_ident      oktidt
#define task_start      oktsta
#define task_restart    oktrst
#define task_suspend    oktsus
#define task_resume     oktrsm
#define task_set_priority oktspr
#define task_set_mode    oktsmd
#define task_read_note_pad oktrnp
#define task_write_note_pad oktwnp
#define task_info       oktinf
```

**/\* Region Operations \*/**

**#ifdef \_\_ANSI\_\_**

```
extern int okrcre( char *name, void *addr, int length, int granularity,  
                  bit_field options, region_id *rid ) ;  
extern int okrdel( region_id rid ) ;  
extern int okridt( char *name, region_id *rid ) ;  
extern int okrgsg( region_id rid, int seg_size, void **seg_addr ) ;  
extern int okrrsg( region_id rid, void *seg_addr ) ;  
extern int okrintf( region_id rid, int size, int max_segment,  
                   int granularity, bit_field options)
```

**#else**

```
extern int okrcre( ) ;  
extern int okrdel( ) ;  
extern int okridt( ) ;  
extern int okrgsg( ) ;  
extern int okrrsg( ) ;  
extern int okrintf( ) ;
```

**#endif**

```
#define region_create      okrcre  
#define region_delete     okrdel  
#define region_ident      okridt  
#define region_get_seg    okrgsg  
#define region_ret_set    okrrsg  
#define region_info       okrintf
```

**/\* Pool Operations \*/**

```
#ifdef __ANSI__
extern int okpcre( char *name, void *addr, int length, int block_size,
                  bit_field options, pool_id *pid ) ;
extern int okpdel( pool_id pid ) ;
extern int okpidt( char *name, node_id nid, pool_id *pid);
extern int okpgbl( pool_id pid, void **blk_addr ) ;
extern int okprbl( pool_id pid, void *blk_addr ) ;
extern int okpinf( pool_id pid, int buffers, int free_buffers,
                  int buff_size, bit_field options)
#else
extern int okpcre( ) ;
extern int okpdel( ) ;
extern int okpidt( ) ;
extern int okpgbl( ) ;
extern int okprbl( ) ;
extern int okpinf( ) ;
#endif

#define pool_create      okpcre
#define pool_delete     okpdel
#define pool_ident      okpidt
#define pool_get_blk    okpgbl
#define pool_ret_blk    okprbl
#define pool_info       okpinf
```

**/\* Semaphore Operations \*/**

```
#ifdef __ANSI__

extern int okscre( char *name, int init_count, bit_field options, sem_id
                  *sid ) ;
extern int oksdel( sem_id *sid ) ;
extern int oksidt( char *name, node_id nid, sem_id *sid ) ;
extern int okstak( sem_id *sid, bit_field options, int time_out ) ;
extern int okssig( sem_id *sid ) ;
extern int oksinf( sem_id *sid, bit_field options, int count,
                  int tasks_waiting)

#else

extern int okscre( ) ;
extern int oksdel( ) ;
extern int oksidt( ) ;
extern int okstak( ) ;
extern int okssig( ) ;
extern int oksinf( ) ;

#endif

#define sem_create okscre
#define sem_delete oksdel
#define sem_ident oksidt
#define sem_take okstak
#define sem_signal okssig
#define sem_info oksinf
```

/\* Queue Operations \*/

#ifdef \_\_ANSI\_\_

```
extern int okqcre( char *name, int max_buff, int length,
                  bit_field options, queue_id *qid ) ;
extern int okqdel( queue_id qid ) ;
extern int okqidt( char *name, node_id nid, queue_id *qid ) ;
extern int okqsnd( queue_id qid, void *msg_buff, int msg_length ) ;
extern int okqjmp( queue_id qid, void *msg_buff, int msg_length,
                  int *count ) ;
extern int okqrcv( queue_id qid, void *msg_buff, int buff_length,
                  bit-field options, int time_out, int length ) ;
extern int okqflu( queue_id qid, int *count ) ;
extern int okqinf( queue_id qid, int max_buff, int length,
                  bit_field options, int messages_waiting,
                  int tasks_waiting)
```

#else

```
extern int okqcre( ) ;
extern int okqdel( ) ;
extern int okqidt( ) ;
extern int okqsnd( ) ;
extern int okqbro( ) ;
extern int okqjmp( ) ;
extern int okqrcv( ) ;
extern int okqflu( ) ;
extern int okqinf( ) ;
```

#endif

```
#define queue_create      okqcre
#define queue_delete     okqdel
#define queue_ident      okqidt
#define queue_send       okqsnd
#define queue_broadcast   okqbro
#define queue_jump       okqjmp
#define queue_receive     okqrcv
#define queue_flush      okqflu
#define queue_info       okqinf
```