**/\* Event Operations \*/**

#ifdef \_\_ANSI\_\_

extern int okesnd( task\_id tid, bit\_\_field event ) ;

extern int okercv( bit\_field event, bit\_field options, int time\_out,

 bit\_field \*event\_received ) ;

#else

extern int okesnd( ) ;

extern int okercv( ) ;

#endif

**#define event\_send okesnd**

**#define event\_receive okercv**

**/\* Exception Operations \*/**

#ifdef \_\_ANSI\_\_

extern int okxcat( int bit\_number, void new\_xsr(bit\_field), bit\_field

new\_mode, void (\*old\_xsr) (bit\_field), bit\_field

\*old\_mode );

extern int okxrse( task\_id tid, bit\_field exception ) ;

extern void okxret( void ) ;

#else

extern int okxcat( ) ;

extern int okxrse( ) ;

extern void okxret( ) ;

#endif

#define exception\_catch okxcat

#define exception\_raise okxrse

#define exception\_return okxret

**/\* Clock Operations \*/**

#ifdef \_\_ANSI\_\_

extern int okcset( clock\_buff \*clock );

extern int okcget( clock\_buff \*clock )

extern int okctik( void ) ;

#else

extern int okcset( );

extern int okcget( );

extern int okctik( );

#endif

#define clock\_set okcset

#define clock\_set okcget

#define clock\_tick okctik

**/\* Timer Operations \*/**

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

extern int oktmwa( int ticks ) ;

extern int oktmww( clock\_buff \*clock ) ;

extern int oktmea( int ticks, bit\_fie1devent, timer\_\_id \*tmid ; extern int oktmew( clock buff \*clock, bit\_field event, timer\_id \*tmid ( );

extern int oktmee( int ticks, bit\_field event, timer\_\_id \*tid );

extern int oktmca( timer\_id \*tmid\_) ;

#else

extern int oktmwa( );

extern int oktmww( );

extern int oktmea( );

extern int oktmew( );

extern int oktmee( );

extern int oktmca( );

#endif

#define timer\_wake\_after oktmwa

#define timer\_wake\_when oktmww

#define timer event after oktmea

#define timer:event:when oktmew

#define timer\_event\_every oktmee

#define timer\_cancel oktmca

**/\* Interrupt Operations \*/**

#ifdef ANSI

extern int okient( void ) ;

extern void okiret( void ) ;

**#else**

extern int okient( )

; extern void okiret( ) ;

#endif

#define int\_enter okient

#define int\_return okiret

/\*

**COMPLETION STATUS CONSTANTS**

This section of the ORKID C language binding contains definitions for

all the completion status values used in the main ORKID standard. The symbols used are the same as those given in the main standard, and are defined for C by this standard. \*/

#define OK ???

#define CLOCK NOT SET ???

#define ILLEGAL\_USE ???

#define INVALID\_ARGUMENT ???

#define INVALID\_BIT ???

#define INVALID\_BUFF ???

#define INVALID\_BUFF\_SIZE ???

#define INVALID\_CLOCK ???

#define INVALID\_COUNT ???

#define INVALID-GRANULARITY ???

#define INVALID-ID ???

#define INVALID\_LENGTH ???

#define INVALID\_LOCATION ???

#define INVALID-NODE ???

#define INVALID\_OPTIONS ???

#define INVALID\_PARAMETER ???

#define INVALID\_PRIORITY ???

#define INVALID\_SEGEMENT ???

#define NAME\_NOT\_FOUND ???

#define NODE\_NOT\_REACHABLE ???

#define NO\_EVENT ???

#define NO\_MORE\_MEMORY ???

#define OBJECT\_DELETED ???

#define OBJECT\_NOT\_LOCAL ???

#define OBJECT\_PROTECTED ???

#define POOL\_IN\_USE ???

#define POOL\_NOT\_SHARED ???

#define POOL\_OVERLAP ???

#define QUEUE\_DELETED ???

#define QUEUE\_EMPTY ???

#define QUEUE\_FULL ???

#define REGION\_IN\_USE ???

#define REGION\_OVERLAP ???

#define SEMAPHORE\_DELETED ???

#define SEMAPHORE\_NOT\_AVAILABLE ???

#define SEMAPHORE\_OVERFLOW ???

#define SEMAPHORE\_UNDERFLOW ???

#define TASK-ALREADY\_STARTED ???

#define TASK\_ALREADY-SUSPENDED ???

#define TASK\_NOT\_STARTED ???

#define TASK\_NOT\_SUSPENDED ???

#define TIME\_OUT ???

#define TOO\_MANY\_OBJECTS ???

#define XSR\_NOT\_SET ???

/

**Literal Values**

This section of the Orkid C language binding contains definitions for all special symbols used as argument values in the main ORKID standard. The symbols used are the same as those given in the main standard, and are defined for C by this standard. \*/

#define LOCAL\_NODE ??? /\* nid \*/

#define OTHER\_NODES ???

#define ALL\_NODES ???

#define WHO\_AM\_I ??? /\* name \*/

#define SELF ??? /\* tid \*/

#define RUNNING ??? /\* state \*/

#define READY ???

#define BLOCKED ???

#define SUSPENDED ???

#define CURRENT ??? /\* new\_prio \*/

#define HIGHP ??? /\* new\_prio, old\_prio \*/

#define NOXSR ??? /\* new\_mode, mode, mask, old\_mode \*/

#define NOTERMINATION ???

#define NOPREEMPT ???

#define NOINTERRUPT ???

#define ALL ??? /\* mask \*/

#define GLOBAL ??? /\* options \*/

#define FORCED\_DELETE ???

#define FIFO ???

#define ANY ???

#define NOWAIT ???

#define URGENT ???

#define ZERO ??? /\* options, mask, modes \*/

#define FOREVER ??? /\* time\_out \*/

#define NULL\_XSR ??? /\* new\_xsr, old\_xsr \*/

#endif