



EMBEDDED SOFTWARE DEVELOPMENT TOOLS

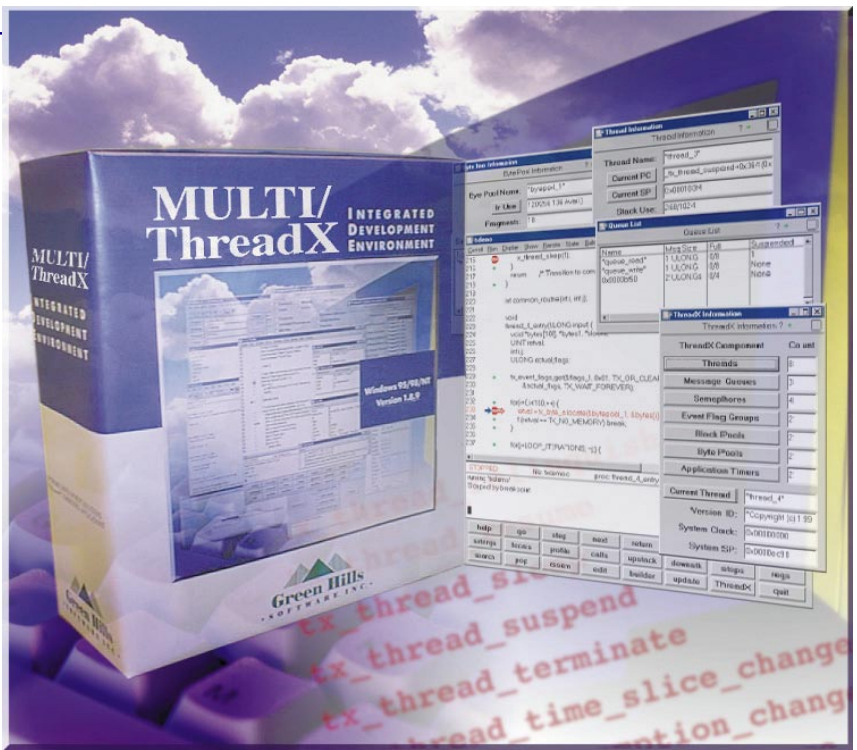
THREADX® FOR STRONGARM®

THREADX RTOS

The ThreadX Real-Time Operating System is a highly efficient, robust, royalty-free kernel designed for deeply embedded applications requiring a small footprint and rapid real-time response. ThreadX provides super-fast context switching times while giving developers a rich set of services to use for task control and communication. ThreadX is fully integrated with Green Hills Software's MULTI® Integrated Development Environment, providing Optimizing Compilers, Source Debugger, Graphical Project Builder and many other powerful tools for development of embedded applications.

HIGHLIGHTS

- Complete StrongARM support
- Reasonable pricing
- No Royalties
- Complete ANSI C source code
- Easy to use and powerful services
- Responsive Technical Support
- Unlimited Threads, Queues, Event Flags, Timers, Semaphores, Block Pools, and Byte Pools
- Flexible memory usage
- Timeout available on all thread suspension
- Advanced *preemption-threshold* technique
- Low-overhead Application Timers
- Size scales automatically
- *picokernel* architecture for size and speed
- Small footprint (sizes in bytes)
 - Instruction area size: 4-25K
 - Global data area: 696
 - Thread control block: 144
 - Timer control block: 44
 - Semaphore control block: 28
 - Queue control block: 56
 - Event Flag control block: 32
 - Block Memory control block: 48
 - Byte Memory control block: 52



- Fast Execution (20MHz, 0 wait-states, StrongARM)

context switch	3.5us
tx_thread_suspend	10.0us
tx_thread_resume	8.0us
tx_thread_relinquish	4.0us
tx_semaphore_get	3.0us
tx_semaphore_put	3.5us
tx_queue_send	6.5us
tx_queue_receive	5.0us
tx_event_flags_set	4.0us
tx_event_flags_get	4.5us
tx_block_allocate	3.5us
tx_block_release	4.0us
tx_byte_allocate	10.0us
tx_byte_release	5.0us

STRONGARM OPTIMIZATIONS

ThreadX optimizes context switching on the StrongARM. When context switching occurs inside of a ThreadX service call, only the registers preserved across function calls are saved as part of the thread's context, i.e. registers v1-v7, fp, and lr.

A similar technique is used in interrupt handling. On the front end of interrupt service routines, only the

compiler's scratch registers are saved initially (registers a1-a4). The full register set is saved only if thread preemption is required.

FAST FIQ INTERRUPT RESPONSE

FIQ interrupts are left completely enabled throughout ThreadX processing, resulting in the fastest possible response.

IMPROVE YOUR STRONGARM DEVELOPMENT

Let our extensive experience with the StrongARM family of microprocessors help your product development. The ThreadX high-performance real-time kernel, helps improve your product's quality and its time-to-market. In addition, using ThreadX makes it easier to enhance your product in the future.



THREADX FOR STRONGARM