expresslogic

When It Really Counts!

express**logic**

Express Logic, Inc., is a San Diego, CA based developer of real-time operating systems (RTOS) and related products for embedded applications. Founded in 1996, Express Logic has consistently led the industry with high-value software products and responsive, reliable customer support.

Express Logic was founded by William E. (Bill) Lamie and David L. Lamie. Self-funded and privately operated since its inception, Express Logic operates from its San Diego, CA headquarters, with offices and representatives worldwide.

Express Logic's products include the popular ThreadX® real-time operating system (RTOS), NetX™ TCP/IP stack, FileX® embedded file system, USBX™ host/device USB stack, PegX™ an embedded GUI development kit, StackX™ stack size analyzer, and TraceX® graphical real-time event trace utility.

◄ Mobile Deployments

Express Logic's ThreadX® RTOS has been deployed in over **1 Billion** electronic products, spanning a variety of markets. Among those are over 500 million mobile devices, a milestone recently recognized by VisionMobile in their semi-annual survey of mobile device deployments. VisionMobile, a market analysis and strategy firm delivering market know-how to the mobile industry, offers research reports, industry maps, training courses and advisory services on emerging technologies.

"We welcome Express Logic into the 100 Million Club based on the widespread use of their ThreadX RTOS product in mobile devices," commented Andreas Constantinou, Research Director of VisionMobile. *"ThreadX already has amassed over 500 million mobile deployments and represents one of a small number of software products that can claim such success."*

Fast Time to Market ▶

Another survey, conducted annually by Embedded Market Forecasters, shows that developers who used ThreadX reported that they completed their development projects on-time or ahead of schedule more often than those using other RTOSes – including Linux – over 70% of the time! Use ThreadX and get YOUR next product to market faster.
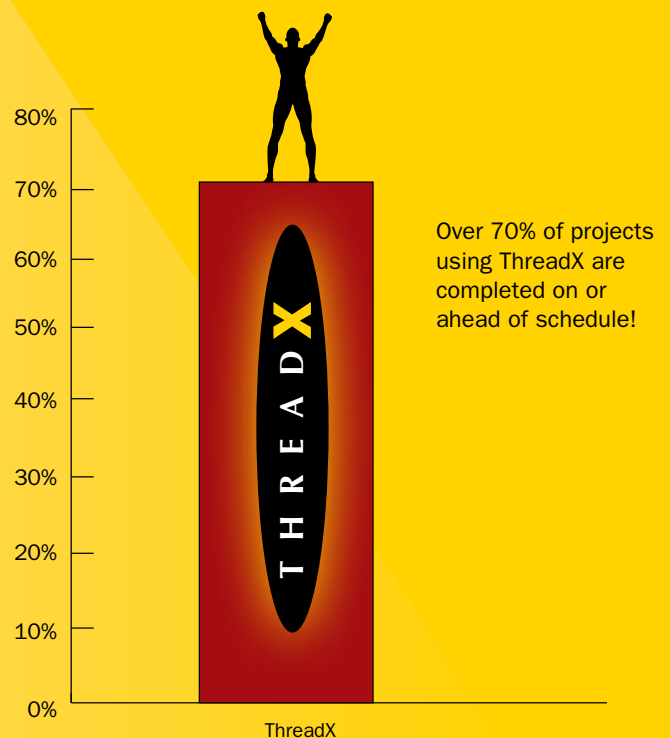
Get Our Newsletter

Express Logic publishes informative Newsletters with the latest product introductions, technology spotlight, and related industry news. To receive this free newsletter, visit www.rtos.com/downloads, and look for the link to Opt-In.

Over 70% of projects using ThreadX are completed on or ahead of schedule!

80%
70%
60%
50%
40%
30%
20%
10%
0%

THREADX

ThreadX

Who is using our Products?

Markets Served

It's no surprise that over 1 billion embedded devices have been deployed using ThreadX, and more are on the way at a rate of over 200 million per year! These devices are found in high-volume market areas such as consumer electronics, medical devices, and industrial automation equipment. Our web site has a sampling of customer testimonials from each of these areas.

- **Consumer Devices.** With hundreds of million consumer devices powered by Express Logic's ThreadX RTOS, NetX TCP/IP networking stack, and FileX file system, Express Logic outpaces all other suppliers to consumer device development.

- **Medical Devices.** From hand-held diagnostic equipment to respirators and ventilators, Express Logic products meet the needs of medical device manufacturers, and the requirements of FDA certification.

- **Networking.** From wireless, hand-held devices to cell phones, to DSL routers, Express Logic's products are ideal for high-performance data networking applications.

- **System-On-Chip (SoC)** manufacturers increasingly turn to Express Logic's ThreadX for its small, fast and royalty-free benefits. Express Logic enables SoC development with the fastest time-to-market and lowest possible cost.

- **Industrial Automation/Automotive.** Express Logic delivers cost-effective solutions for industrial automation and automotive applications.

- **Aerospace/Defense.** Developers employ Express Logic's products because of their small memory footprint, ease of use, and ease of certification for demanding FAA applications.

Representative ThreadX Deployments

Product Category	ThreadX Deployments	Representative Customers
Wireless Networking	750,000,000	Broadcom, Intel, Marvell
Ink-Jet Printers	375,000,000	HP, Sharp
Baseboard Management Controllers	50,000,000	Intel, QLogic
Cell Phones	30,000,000	Samsung, Infineon, Datang
Digital TV	16,000,000	Sony, Pioneer, Zoran
Digital Cameras	15,000,000	HP, Pentax, Zoran
DVD Recorders/Players	7,000,000	Toshiba, Sharp, Zoran
Storage Devices	3,500,000	ST, Quantum
DSL/Cable Modems	3,100,000	Conklin
Medical Devices	2,000,000	Welch-Allyn
Digital Radio	1,000,000	IBiquity
Space Probes	2	NASA

This chart shows an approximate breakdown of more than 1 billion devices into product categories such as wireless devices (bluetooth, baseband radio, WiFi), Printers, Cameras, Digital TV, and the like. You can see the types of products most widely developed using ThreadX, and some representative customers who make those products.

Partners

Express Logic is extremely proud to have many industry leading partners who provide technologies that complement our own products. From development tools, to communications middleware, to microprocessors, and more, Express Logic's partners form an ecosystem that enables developers to choose ThreadX-compatible, integrated, and supported products to complete their needs.

WIND RIVER   MIPS TECHNOLOGIES   Allegro   ATMEL   applied micro
freescale Alliance Member   ARM CONNECTED   IAR SYSTEMS   embWiSe   mindready
RENESAS Everywhere you imagine.   tensilica   ALTERA   OBJECTIVE INTERFACE   ANALOG DEVICES
LAUTERBACH
ENERGY micro   AMCC   aicas realtime   Datalight   XILINX   AMD
Digi   ARC   Oki Semiconductor   ST   CIRRUS LOGIC
TEXAS INSTRUMENTS   MICROCHIP   CEVA   NXP

Hewlett Packard has licensed the use of ThreadX for all Inkjet and all-in-one devices.

Zoran uses ThreadX in its SoCs for digital cameras, printing, television, and other consumer electronic devices.

ThreadX is widely used in networking controlling Bluetooth and Baseband Radios in cell phones.

Welch Allyn uses ThreadX in a wide range of its medical equipment, including defibrillators, blood analyzers, and monitors.

Pulmonetic Systems uses ThreadX in ventilators that keep paralyzed patients supplied with life-critical oxygen.

ThreadX is used in the automotive industry in a number of applications, including test equipment and digital radios.

In November, 2006, the Mars Reconnaissance Orbiter (MRO) entered an orbit around Mars and began taking spectacular high-resolution images of the Red Planet.

Business Model

Express Logic's business model is based on a royalty-free approach. We do not charge per-unit fees for deployed units. Instead, we charge for licenses that authorize the use of ThreadX in the development and production of individual, or a family of products. In all cases, full ThreadX source code is provided, and direct telephone, e-mail, and in-person support is available for an annual fee.

THREAD X

ThreadX is a real-time operating system (RTOS) designed for embedded, real-time applications running on microcontrollers, microprocessors, or DSPs.

THREADX Snapshot

- Small-footprint (small as 2KB, automatic scaling)
- Fast execution (sub microsecond context switch)
- Supports all popular processors and tools (see below)
- TraceX system analysis support
- Advanced technology:
  - Preemption-Threshold™
  - Event Chaining™
  - Performance metrics
  - Execution Profiling
  - Run-time and Static Stack Analysis
  - Multicore support (SMP and AMP)
  - Downloadable Application Modules
  - Memory Protected Modules
- Extensive ThreadX ecosystem
- Safety-critical Certification Pack™
- Deployed in over 1 Billion devices
- Full Source Code
- Royalty-Free

Fast Time To Market

ThreadX is easy to install, learn, use, debug, verify, certify and maintain, for multiple reasons:

1. **Source Code Availability -** including kernel
2. **Easy-to-use API -** speeds programming and minimizes errors
3. **Quickly Migrate Legacy Code -** POSIX, uITRON, OSEK APIs
4. **Broad 3rd Party Tools Integration -** IAR Embedded Workbench™, GNU, ARM RealView®, Green Hills MULTI®, Wind River Workbench™, MIPS SDE, Renesas HEW, Metaware SeeCode™, Freescale CodeWarrior, Lauterbach TRACE32®, TI CodeComposer Studio, Analog Devices Visual C++, Xilinx EDK, Xtensa, and Altera.

Low Cost

ThreadX is licensed royalty-free, for the lowest possible manufacturing cost. ThreadX is Field Proven, lowering support costs. Proven reliability in over 1 billion deployments dramatically lowers support costs.

Small Size

ThreadX is small in size, as small as 2KB ROM, so it can fit within even the most demanding memory size constraints imaginable. Only those services used by the application are linked into the final image.

High Performance

ThreadX delivers the highest performance found in commercial real-time operating systems, significantly faster than other RTOSes.

- **Fast Boot.** ThreadX boots in less than 120 cycles
- **Optional Removal of basic error checking.** Basic ThreadX error checking can be skipped at a compile-time
- **Picokernel™ Design**, with services not layered on each other
- **Optimized Interrupt Processing.** Only scratch registers are saved/restored upon ISR entry/exit, unless preemption is necessary.
- **Fast Interrupt Response Time, Fast Context Switching, Low RTOS Service Overhead.**

ThreadX Service	Service Time	ThreadX Services	Typical Size in Bytes
Thread Suspend	0.6us	Core Services (required)	2,000
Thread Resume	0.6us	Queue Services	900
Queue Send	0.3us	Event Flag Services	900
Queue Receive	0.3us	Semaphore Services	450
Get Semaphore	0.2us	Mutex Services	1,200
Put Semaphore	0.2us	Block Memory Services	550
Context Switch	0.4us	Byte Memory Services	900
Interrupt Response	0.0us - 0.6us		

Based on 200MHz processor

Memory Protection

ThreadX enables application threads and the ThreadX kernel to be protected against accidental read or write access from other threads. This prevents code or data corruption from latent application bugs, and eliminates one of the most common causes of application crashes.

Downloadable Application Modules

ThreadX enables one or more application threads to be bundled into a "Module" that can be dynamically loaded and run on the target. Modules enable field upgrade, bug fixing, and program partitioning to allow large applications to occupy only the memory needed by active threads.

expresslogic

ThreadX Safety-Critical Certification Pack™

The ThreadX Safety-Critical Certification Pack is a turnkey solution for meeting the RTOS-related aspects of standards for safety-critical systems. The ThreadX Safety-Critical Certification Pack includes all source code, documentation, coverage analysis, validation artifacts, and summary report for submission to the governing agency as part of the overall system certification. A typical ThreadX Safety-Critical Certification Pack contains the following:

- Software Safety Requirements
- Software Safety Validation Plan
- Development Plan
- Configuration Management Plan
- Quality Assurance Plan
- Verification/Test Plan
- Coding Standards
- Requirements Standards
- Design Standards
- Requirements Specification

- Design Description
- Unit Test Procedure
- Unit Test Plans
- Unit Test Reports
- Integration Test Procedure
- Integration Test Plan
- Integration Test Report
- Trace Matrices
- Configuration Index
- Software Accomplishments Summary
- Safety Manual

Avionics:
- RTCA DO-178B
- EUROCAE ED-12B A/B/C/D
- EUROCAE ED109 Level AL1 – AL5

Industrial/Transportation/Rail:
- IEC-61508 SIL 3/2/1
- CENELEC EN5

Medical:
- FDA 510(k)
- IEC-62304 Class A, B, C

Broad Microprocessor Support

ThreadX runs on most popular 32/64-bit microprocessors, out-of-the-box, fully tested and fully supported:

Advanced Architectures: A2P

Altera: NIOS II

AMCC: PowerPC

Analog Devices: SHARC, Blackfin

ARC: ARC 600, 700

ARM: ARM7, ARM9, ARM11, Cortex-M0/M3/M4/R4/R5/A8/A9

Atmel: AVR32, ARM7, ARM9, Cortex-M3

CEVA: CEVA-TeakLite-III

EnergyMicro: EFM32

Freescale: PowerPC, 68K, i.MX, ColdFire, Kinetis,

Intel: x86/Pentium, XScale

Microchip: PIC24/PIC32

MIPS: MIPS32, MIPS64, 4k, 24k, 34k, 1004k

NXP: LPC, ARM7, ARM9, Cortex-M3

Renesas: SH, H8, V850, RX

ST: STM32, ARM7, ARM9, Cortex-M3

Tensilica: Xtensa, Diamond

TI: C5xxx, C6xxx, Stellaris, Sitara

Xilinx: MicroBlaze, PowerPC405

Complete Win32 Simulation is also available.

Execution Profile Kit

The ThreadX Execution Profile Kit (EPK) provides an infrastructure for applications to dynamically track execution time for threads, Interrupt Service Routines (ISRs), and idle system conditions. This is especially useful for debugging and tuning the application for maximum performance.

Percent of Processor Utilization
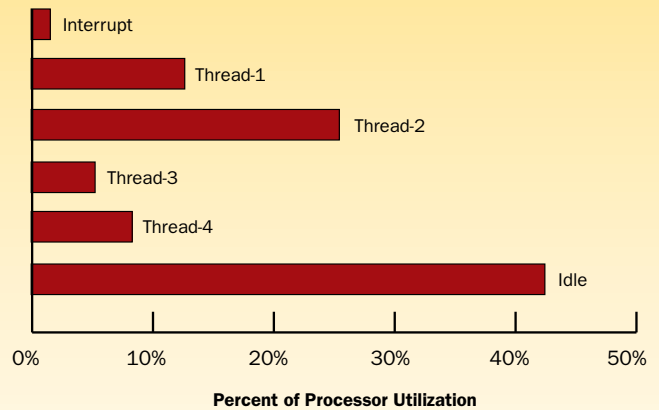
Advanced Technology

ThreadX provides thread-management services with advanced technology for low overhead and ease of use.

- Complete Multitasking Facilities
  - Threads, Application Timers, Message Queues, Counting Semaphores, Mutexes, Event Flags, Block and Byte Memory Pools
- Priority-based preemptive scheduling
- Preemption-threshold™ - Unique to ThreadX, helps reduce context switches and help guarantee schedulability.
- Deterministic
- Event Trace - Capture last "n" system/application events
- Event Chaining™ - Register an application-specific "notify" callback function for each ThreadX communication or synchronization object
- Downloadable Application Modules
- Memory Protection
- Run-Time Performance Metrics
  - Number of thread resumptions
  - Number of thread suspensions
  - Number of solicited thread preemptions
  - Number of asynchronous thread interrupt preemptions
  - Number of thread priority inversions
  - Number of thread relinquishes
  - And more ….
- Execution Profile Kit
- Separate Interrupt Stack
- Run-time Stack Analysis
- Optimized Timer Interrupt Processing
- Priority Flexibility - Up to 1024 priority levels

expresslogic

N E T X

NetX™ and NetX Duo™

NetX and NetX Duo are Express Logic's high-performance IPv4 and dual-stack IPv4/IPv6 implementations of TCP/IP protocol standards. Aside from NetX Duo's IPv6 support, both products are equivalent, and we use "NetX" here to refer to both.

NetX Snapshot

- Small-footprint (small as 5KB, automatic scaling)
- Fast execution (achieve wire speed, no packet copying)
- Supports all popular processors and tools (see below)
- Phase-II IPv6 Ready Logo Certification
- TraceX system analysis support
- BSD-compatible socket layer
- Advanced Features:
  - UDP Fast-Path Technology(TM)
  - Flexible packet management
  - Additional protocols: AutoIP, DHCP, DNS, FTP HTTP, NAT, POP3, PPP, SMTP, SNMP v1-2-3, SNTP, Telnet, TFTP
- Full Source Code
- Royalty-Free

Small Footprint - NetX is implemented as a C library. Only the features used by the application are brought into the final image. The minimal footprint of NetX is as small as 5KB on RISC processors.

Piconet™ Architecture - Many TCP/IP implementations have a considerable number of function calls sprinkled throughout the packet processing path. The elimination of layered function calls and loosely coupled protocol components is what we call a piconet architecture.

Fast Response - With its non-layering Piconet™ architecture and zero-copy API, NetX's packet processing speed is second to none.

Easy To Use - The NetX API is both intuitive and highly functional. Component names are made of real words and not the "alphabet soup" names that are common to other TCP/IP products.

Flexible Packet Pool Management - The application may create any number of packet pools in any number of memory areas.

UDP Fast Path™ Technology - Basic UDP packets pass through NetX without any copying and without any system context switches.

NetX Protocol Components - NetX provides a complete set of protocol components that comprise the TCP/IP standard, including the following:

- Transmission Control Protocol (TCP)
- Internet Protocol (IP)
- User Datagram Protocol (UDP)
- Address Resolution Protocol (ARP)
- Reverse Address Resolution Protocol (RARP)
- Internet Control Message Protocol (ICMP)
- Internet Group Management Protocol (IGMP)

Zero-Copy API - NetX provides zero-copy TCP/IP communication to eliminate processor cycles consumed by moving data to and from user buffers. Many commercial network stacks process all packets received within a system thread thereby adding a context switch.

NetX Duo Phase-II IPv6 Ready Logo Certification

NetX Duo has achieved IPv6-Ready Logo certification, evidence that it has passed conformance and interoperability tests, administered and validated by the IPv6 Forum. There are two phases of testing, Phase-I and Phase-II. Phase-II is much more rigorous and extensive, and is a superset of Phase-I.

IPsec - NetX Duo (optionally) incorporates IPsec (Internet Protocol Security), a protocol suite for securing Internet Protocol (IP) communications by authenticating and encrypting each IP packet of a communication session. IPsec also includes protocols for establishing mutual authentication between agents at the beginning of the session and negotiation of cryptographic keys to be used during the session. IPsec protects any application traffic across an IP network. Applications do not need to be specifically designed to use IPsec.

> " Here's what Sony engineers have said about NetX:
>
> *"On this single SPE, we operated the protocol stack NetX, developed for embedded applications, and the microkernel ThreadX required for its operation. Both were manufactured by Express Logic.*
>
> *As a result, we achieved a TCP performance of 8.5 Gbps for 3-KB packet sizes using a single SPE operating at 3.2 GHz.*
>
> *This result indicates a competitive network protocol processing performance, considering that we employed a processor designed for a variety of computational applications rather than a dedicated network processor, and demonstrates the potential of the application of SPEs in this field."*
>
> – ***Network Processing on an SPE Core in Cell Broadband Engine™***, *Sony Corporation, 16th IEEE Symposium on High Performance Interconnects.* http://www.hoti.org/archive/2008papers/2008_S4_3.pdf "

expresslogic

F I L E X

FileX

FileX is Express Logic's high-performance FAT- compatible file system. It is fully integrated with ThreadX and is available for all supported processors. Like ThreadX, FileX is designed to have a small footprint and high-performance, making it ideal for today's deeply embedded applications.

FILEX Snapshot

- Small-footprint (small as 6KB, automatic scaling)
- Fast execution (user configurable sector caches)
- Supports all popular processors and tools (see below)
- TraceX system analysis support
- Advanced Features:
    - FAT 12/16/32 support
    - Long filename support
    - performance metrics
- Full Source Code
- Royalty-Free

Small Footprint - FileX is implemented as a C library. Only the features used by the application are brought into the final image. The minimal footprint of FileX is as small as 6KB on CISC processors.

Fast Response - FileX has minimal function call layering, an internal logical sector cache, contiguous cluster allocation, and consecutive cluster reading and writing . All of these attributes make FileX extremely fast!

Easy To Use - FileX is very easy to use. The FileX API is both intuitive and highly functional. In addition, the service names are made of real words and not the "alphabet soup" names that are common to other file system products. Building a FileX application is also easy. Simply include the fx_api.h file in the application software and link the application objects with the FileX library.

Easy to Integrate - FileX is easily integrated with virtually any FLASH or physical media device

No Mysteries - FileX is delivered with complete C source code so you can see exactly how it operates. If you are used to in-house file systems or had bad experiences with "black box" commercial products, this should be most comforting.

Advanced Features

FileX provides many advanced features for embedded file applications, including the following:

- 12-, 16-, and 32-bit FAT support
- Long filename support
- Internal FAT entry cache
- Contiguous file allocation
- Consecutive sector and cluster read/write
- Internal logical sector cache

FileX supports an unlimited number of media devices at the same time, including RAM disks, FLASH managers, and actual physical devices. It also supports 12-, 16-, and 32-bit File Allocation Table(FAT) formats, contiguous file allocation, and it is highly optimized for both size and performance. Designed to meet the growing need for FLASH devices, FileX uses the same design and coding methods as ThreadX. Like all Express Logic products, FileX is distributed with full ANSI C source code, and it has no run-time royalties.

Product Highlights

- No royalties
- Complete ANSI C source code
- Real-time performance
- Responsive technical support
- Unlimited FileX objects (media, directories, and files)
- Dynamic FileX object creation/deletion
- Flexible memory usage
- Size scales automatically
- Small footprint (as low as 6 KBytes) instruction area size: 6-30K
- Complete integration with ThreadX
- Endian neutral
- Easy-to-implement FileX I/O drivers
- RAM disk demonstration runs out-of-the-box
- Media format capability
- Error detection and recovery
- Fault tolerant options
- Built-in performance statistics

expresslogic

USBX

USBX is Express Logic's high-performance embedded USB host/device/OTG stack. It is fully integrated with ThreadX and is available for all ThreadX supported processors. Like ThreadX, USBX is designed to have a small footprint and high-performance, making it ideal for today's deeply embedded applications that interface with USB devices or are connected with a Windows/Apple/Linux desktop over USB.

USBX Snapshot

- Small-footprint (small as 10KB, automatic scaling)
- Fast execution
- Supports all popular processors and tools (see below)
- Passed USB-IF USBCV Test
- Passed OTG Protocol Test (OPT)
- TraceX system analysis support
- Host, Device, and OTG support
- Extensive Device Class support:
   CDC, HID, PIMA, RNDIS, STORAGE
- Extensive Host Class support:
   ASIX, AUDIO, CDC, HID, HUB, PIMA,
   PRINTER, PROLIFIC, STORAGE
- Full Source Code
- Royalty-Free

Small Footprint - USBX is implemented as a C library. Only the features used by the application are brought into the final image. The minimal footprint of USBX is under 10KB.

Easy To Use - USBX is very easy to use. The USB API is both intuitive and highly functional. In addition, the service names are made of real words and not the "alphabet soup" names that are common to other USB products. Building a USBX application is also easy. Simply include the ux_api.h fill in the application software and link the application objects with the USBX library.

Powerful Services of USBX

Multiple Host Controller Support

USBX can support multiple USB host controllers running concurrently. This feature allows USBX to support the USB 2.0 standard using the backward compatibility scheme associated with most USB 2.0 host controllers on the market today.

USB Software Scheduler

USBX contains a USB software scheduler necessary to support USB controllers that do not have hardware list processing. The USBX software scheduler will organize USB transfers with the correct frequency of service and priority, and instruct the USB controller to execute each transfer.

Complete USB Device Framework Support

USBX can support the most demanding USB devices, with multiple configurations, multiple interfaces, and multiple alternate settings.

Major Features

- Host/Device/Combined Operation
- USB low speed, full speed and high speed are supported
- Supports all embedded CPUs
- Supports many USB host/device controllers in discrete or IP form integrated with a SOC including Synopsis, Philips, Atmel, PowerPC, ColdFire, STM32, NXP, Renesas SH-2A and V8xx, Microchip PIC32, and ADI Blackfin.
- Supports many standard USB class drivers including Mass Storage, Printer, HID, Asix, Audio, Hub, RNDIS, CDC, Pima 15740 and Pictbridge
- Integrated with Express Logic components (FileX and NetX)

USBX supports the two existing USB specifications: 1.1 and 2.0. It is designed to be scalable and will accommodate simple USB topologies with only one connected device as well as complex topologies with multiple devices and cascading hubs. USBX supports all the data transfer types of the USB protocols: control, bulk, interrupt, and isochronous. USBX supports both the host side and the device side. Each side is comprised of three layers:

- Controller layer
- Stack layer
- Class layer

USBX Host mode - USBX in host mode is used when the application requires communication with USB devices such as a USB keyboard, a USB printer or USB Flash disk.

USBX Device mode - USBX in device mode is used when the application requires communication with a Windows/Apple/Linux desktop. In this case the embedded device is considered to be a USB device or slave.

expresslogic

PEGX

The PEGX™ (Portable Embedded GUI for ThreadX) family of GUI development tools are professional quality graphical user interface packages created to meet the needs of embedded systems developers. PEGX products are small, fast, and can be used with any hardware configuration capable of supporting graphical output. PEGX also delivers exceptional visual appeal and an intuitive and powerful API for application-level user interface development. PEGX software accelerates GUI design for embedded devices by allowing developers to create prototypes on a Windows or Linux-based PC.

PEGX Interface - PEG's default appearance is almost identical to Windows. This makes it easy for Windows programmers to get up and running. The diagram shows some example windows, dialogs, and graphic displays created by PEGX.

PEGX Licensing - PEGX is licensed on a per-developed-product basis, eliminating royalty fees. PEGX is delivered with full source code, many example application programs, a complete set of supporting utility programs, hardware interface objects for several common video configurations and input devices, and very thorough documentation to simplify the integration process.

PEGX Features

- Completely customizable
- Multi-lingual support
- High color depth support
- Libraries delivered with full source code
- Small footprint
- Fast execution speed
- Designed for cross platform application development
- Touchscreen support

PEGX Benefits

- Reduce development time and costs
- Rapid user interface development
- Resolve product usability issues before committing to a physical design
- Standardize on graphics software solutions across products
- Differentiate your product with a sophisticated user experience

PEGX Tools

- **PEGX™** - high performance, small footprint GUI tool for monochrome to full color LCDs up to 24 bits per pixel (bpp). PEGX includes PEG WindowBuilder™, has full windowing capability, overlapping windows, simple anti-alias line and font drawing, and full set of widgets/controls.

- **PEGX Pro™** - for today's most sophisticated graphics and high-resolution displays. PEGX Pro is designed for applications needing multiple graphic layers, screen transitions, gradient filled buttons, per pixel alpha blending, BiDi and RichTextBox, and a full set of widgets/controls. Includes PEG WindowBuilder™, and supports 16 bpp to 32 bpp color depths.

- **C/PEGX™** - extremely efficient; designed for small displays, limited memory requirements & lower color depths. Includes C/PEG WindowBuilder, FontCapture and ImageConvert utilities. This visual development environment allows for rapid prototyping of embedded interfaces, validating the design concepts and usability for key stakeholders.

- **PEG WindowBuilder™** - a complete visual layout and design tool to enable GUI design work to take place in parallel to the embedded hardware/software development. In addition — and unlike other graphic design tools — WindowBuilder automatically generates C++ or C source code that is ready to be compiled and linked into the application, further accelerating the deployment of the final product. FontCapture and ImageConvert are now built into the new PEG WindowBuilder for PEGX and PEGX Pro. Custom user-supplied control types, graphics and fonts can easily be incorporated.

expresslogic

TRACEX

TraceX®

TraceX is Express Logic's host-based analysis tool that provides developers with a graphical view of real-time system events and enables them to visualize and better understand the behavior of their real-time systems.

TraceX Snapshot

- Visual system analysis tool for applications using ThreadX, NetX, FileX, and USBX
- Extensive event search and navigation facilities
- Easily measure delta time between events
- Zoom in-out
- Sequential and Time display modes
- Automatically detect priority inversions
- Easily add custom user events
- Built-in performance analysis
- Execution profile, including Idle and ISR time
- Thread stack usage profile
- Performance statistics (context-switches, etc)
- FileX statistics (file read/writes, etc.)
- NetX statistics (packets sent/received, etc.)
- Multicore Support
- Runs on any Windows host
- No license keys

Trace Data Collected By ThreadX® RTOS

TraceX is designed to work with Express Logic's ThreadX RTOS, which constructs a database of system and application "events" on the target system during run-time. These events include:

- Thread context switches
- Preemptions
- Suspensions
- Terminations
- System interrupts
- Application-specific events
- All ThreadX API calls

Events are logged under program control, with time-stamping and active thread identification so they can be displayed later in the proper time sequence, and associated with the appropriate thread. Event logging may be stopped and restarted by the application program dynamically, for example, when an area of interest is encountered.
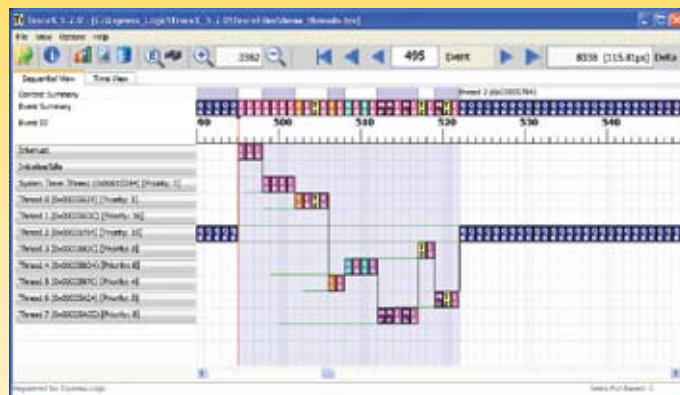
Target-Resident Buffer Holds Event Log Information

Trace information is stored in a circular buffer on the target system, with the buffer location and size determined by the application at run-time. The trace information may be uploaded to the host for analysis at any time – either post mortem or upon a breakpoint.

Sequential Mode

TraceX can operate in Sequential Mode or Time Mode. In Sequential Mode, all events are displayed back-to-back, regardless of the time between them. This enables developers to see all system events, in correct sequence, compressing time into the smallest possible area.

Time Mode

In Time Mode, events are separated along the horizontal axis, according to their real-time occurrence. A time scale appears at the top to indicate the number of clock ticks between events, and the clock tick duration of each thread's run-time.

TraceX is Like a Software Logic Analyzer

Once the event log has been uploaded from target memory to the host, TraceX displays the events graphically on a horizontal axis representing time, with the various application threads and system routines to which the events are related listed along the vertical axis. TraceX creates a "software logic analyzer" on the host, making system events plainly visible. Events are represented by color coded icons, located at the point of occurrence along the horizontal timeline, to the right of the relevant thread or system routine. When an event icon is selected, the corresponding information for that event is displayed. This provides quick, single-click access to the most immediate information about the event and its immediately surrounding events. TraceX provides a "Summary" display that shows all system events on a single horizontal line to simplify analysis of systems with many threads.

expresslogic

StackX™

StackX™ is a unique, patent pending, development tool that helps developers avoid stack overflow problems that traditionally have plagued embedded systems. StackX performs a comprehensive analysis of a complete application at the executable code level, computing the worst-case stack usage the application can experience. Express Logic is the only RTOS company that can identify how large a stack customers need for each application thread and automatically alerts them if they fail to allocate sufficient memory!

StackX SNAPSHOT

- Helps eliminate stack overflow
- Visual analysis of system's call tree and stack
- Supports any ARM ELF file
- Patent pending algorithm
- Extensive search and navigation facilities
- Automatic detection of thread stack overflows
- Detects indirect and recursive calls
- Runs on any Windows host
- Costs less than a single stack overflow!

In the C programming language, the stack—a region of memory on which local variables are created and function arguments are passed—is allocated by the programmer, with the amount based on factors such as machine architecture, OS, application design, and amount of memory available. When the program requires more memory for its stack than has been allocated, the stack overflows—without warning in most cases—which can corrupt other memory areas and typically results in a program crash. Such crashes are very difficult to trace back to the stack overflow, causing programmers to expend considerable time and energy to find the overflow. As a result, they tend to over-allocate stack memory as a precaution.

Deciding how much memory to allocate for the stack has been a trial and error process. As widely respected industry commentator and consultant, Jack Ganssle, has observed:

> "**With experience, one learns the standard, scientific way to compute the proper size for a stack: Pick a size at random and hope**."
> – Jack Ganssle, "The Art of Designing Embedded Systems," Elsevier, 1999.

StackX solves this problem by computing the required stack size automatically, enabling developers to allocate enough memory to be safe, without wasting any. StackX analyzes a program's executable (.elf) file, mapping all function calls, interrupt service routines, local variables, and any other factors related to stack usage. StackX produces a "call-tree" that shows the maximum stack usage for each thread. While it may be used with any .elf file, StackX has particular visibility into ThreadX® RTOS-based applications, with its thread-aware, stack size calculation.

Call Tree Display

There are four types of nodes in the call tree display presented by StackX, namely, the Program Node, Entry Node, Thread Node, and Root Node. The highest-level callers are shown on the left and each function called is shown beneath the node, and indented slightly to the right.
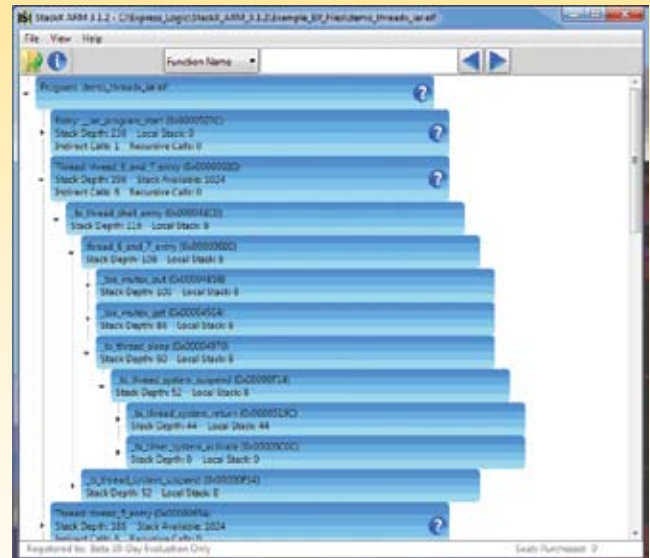
The nodes are displayed in a particular order. The Program Node is always displayed first, followed by the Entry Node. These nodes are always displayed in this order. Thread Nodes are displayed next, in order of their maximum stack usage (from highest to lowest). Note that Thread Nodes are only detected if ThreadX is being used, since StackX knows how ThreadX threads are created and where to find the created stack size. Finally, Root Nodes are displayed, in order of their maximum stack usage. Root Nodes are simply nodes that are not directly called anywhere else. Typically Root Nodes are Interrupt Service Routines (ISRs) or the target of indirect function calls.

The call tree information helps developers understand the stack usage of each function (or thread) and enables them to focus their optimization efforts on these areas.

Additional Program Information

In a system with limited resources, recursive calls must be used with caution. Understanding the call tree in a large and/or complex system is a challenge, not to mention knowing if there is any recursion. StackX helps solve this problem by identifying all recursive function calls. With this knowledge, the developer can determine whether the recursion is necessary and what impact it will have on stack usage.

In summary, StackX provides a perspective of stack usage that is not available in any other tool and at a very low cost - less than the cost of chasing down 1 stack overflow!

expresslogic

express**logic**