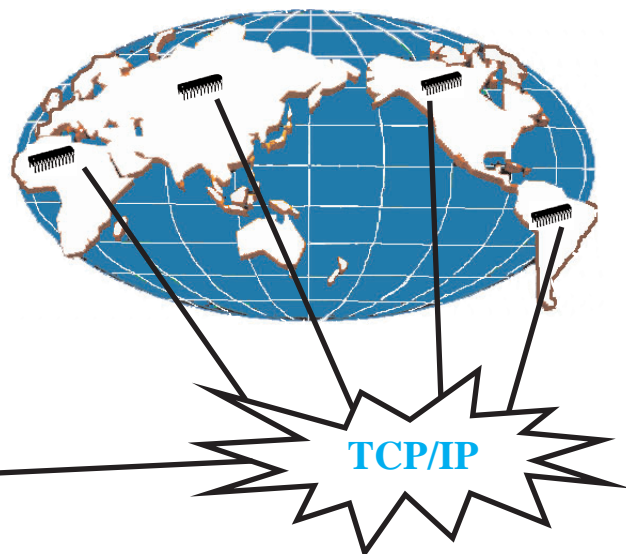
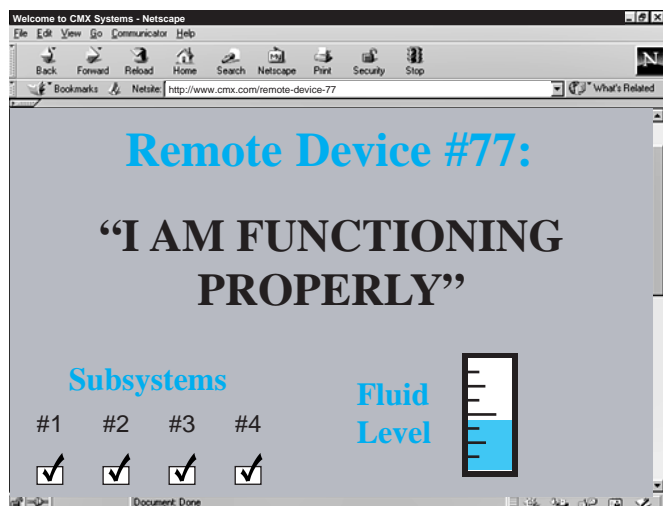




**COMPLETE EMBEDDED SOLUTIONS**

## True TCP/IP Networking on 8-, 16- and 32-Bit Processors!



### Rom Specifications: 5K-32K bytes

*Depending upon processor, protocols used and options selected*

### Finally, the *Right* Connectivity Solution for Your 8-bit, 16-bit and 32-bit Embedded Processors

CMX-MicroNet has been developed by the company that is famous for providing complete, elegant solutions to the embedded community - CMX! Our developers have the expertise and hands-on experience to satisfy the most stringent real time demands that the 8-, 16- and 32-bit community deals with every day. When we set about the task of creating the first true TCP/IP stack for these popular processors, we knew that it had to have:

**Only Industry Standard Protocols.** Of what benefit are closed, proprietary protocols that constrain and confuse your development team? CMX-MicroNet offers only industry standard protocols running right on your target processor and we provide full source code with every sale!

**Use your Current Processor.** Why should you have to upgrade your current processor, or, worst yet, add another processor just for TCP/IP? Those hardware costs can really add up! CMX-MicroNet allows you to work with your current design and still implement the networking connectivity you need.

**An Affordable Pricing Structure.** CMX-MicroNet offers a low, one-time fee and no royalties on deployed products. And you get the entire source code for free with every purchase!

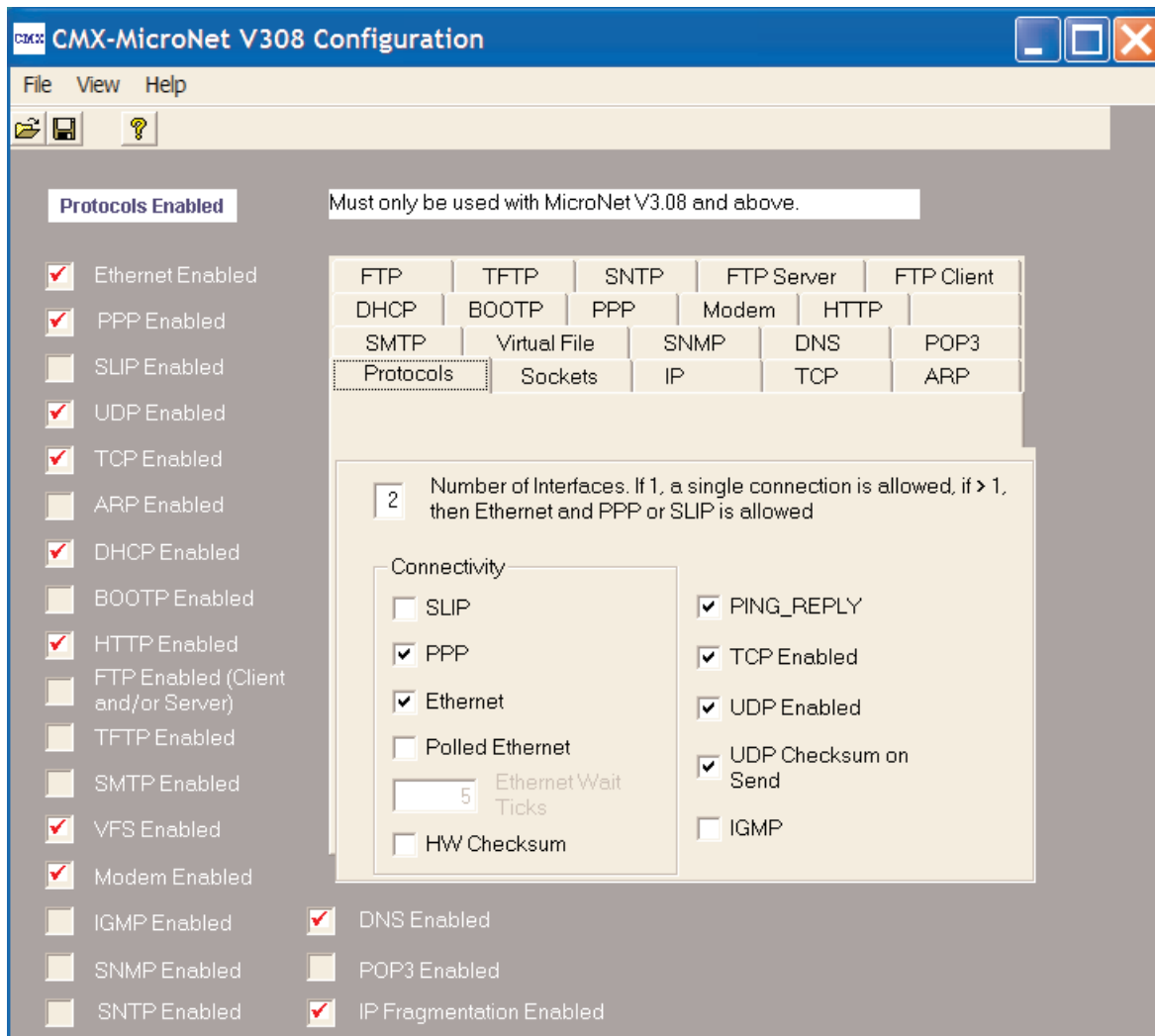
### Supported Protocols

- TCP
- PPP
- UDP
- SLIP
- IP
- HTTP Web Server
- DHCP
- FTP
- TFTP
- SMTP
- SNMP
- POP3
- DNS
- SNMP

### Connectivity

- Ethernet
- Wireless Ethernet
- Dial Up
- Direct

# CMX-Micronet Configuration Manager



*CMX-MicroNet is easy to configure and integrate with your application. Get your embedded processor networked FAST with CMX-MicroNet.*

## *Important Features of CMX-MicroNet*

- **Tested and Proven with Hundreds of Design Wins Around the World**
- **Extremely Small ROM/RAM Requirements**
- **Supports Virtually All 8-, 16-, 32-bit Processors and DSPs**
- **Software Solution does not Require Additional Processor**
- **Web Pages May Contain CGI calls & Server Side Includes**
- **FTP Files, Including New Firmware Send Emails**
- **Can Serve up Java Applets**
- **No Proprietary Protocols**
- **Runs Stand Alone or with any RTOS**
- **Economical One Time Fee, Excellent Manuals and Support**
- **Full Source Code Provided**
- **No Royalties on Shipped Products**

**Contact CMX for a current list of supported processors.**

## **Portability**

CMX-MicroNet has been designed for maximum portability and is written in 100% standard C code. As a consequence, some clients have decided to port the software to processors not yet supported directly by CMX, including microprocessors, microcontrollers and DSPs. Please contact CMX about this option.

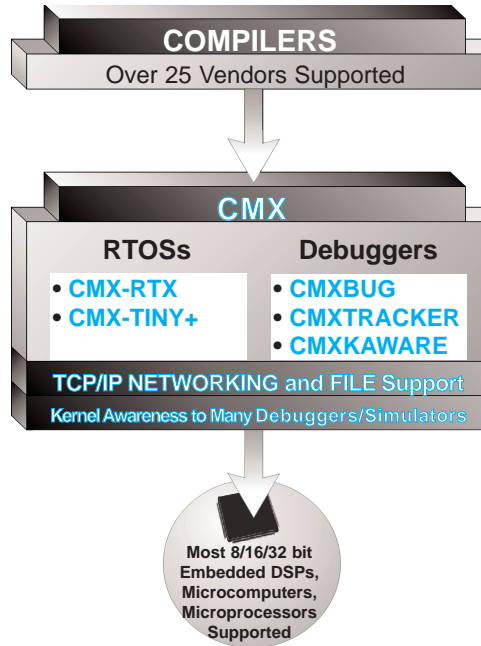
## IS YOUR PROCESSOR IN NEED OF A REAL-TIME MULTI-TASKING OPERATING SYSTEM?

- ◆ Does your processor control the way you program?
- ◆ Do you spend too much time figuring out how to make a section of code execute when it needs to?
- ◆ Do you constantly have to test flags (or go to routines that test) to see if you should execute a certain function?
- ◆ Do you spend too much time with interrupt routines while trying to write the code necessary to process the interrupt's event because the main code would not get to it in a timely manner?

If you answered YES to any of the above questions, you would certainly benefit from considering an RTOS for your application.

In some cases, well structured linear programming is sufficient for a product. In most cases, however, programmers appreciate not having to worry about structuring their code to perform all necessary tasks in a timely manner. This is where CMX-RTX can help. CMX-RTX allows tasks (pieces of code that do specific duties) to run quasi-concurrently. This means that tasks will seem to run all at the same time - doing many specific jobs simultaneously.

CMX-RTX takes the worry and headaches out of real time programming. Our software lets you concentrate on the overall application while taking care of the little details for you. Finish your projects faster and more efficiently with CMX-RTX in your programmer's toolbox!



*CMX-RTX offers a truly preemptive, multi-tasking operating system.*

## WHAT DOES TRULY PREEMPTIVE MEAN?

Some RTOS vendors offer only cooperative scheduling which means that the running task has to call the scheduler to perform a task switch. Others offer time slicing in which each task runs for a certain period of time at which point a task switch takes place no matter what. Other vendors claim to be fully preemptive, yet they do not allow any interrupt to cause a preemption. All of these models will fail you at one point or another.

CMX-RTX allows a task of higher priority that is able to run (whether starting or resuming) to preempt the lower priority running task. This will cause the scheduler to save the context of the running (lower priority) task and restore the context of the higher priority task so that it is now running. A truly preemptive RTOS allows interrupts to cause an immediate task switch. This means that the interrupts now have the added ability of using the RTOS's functions and causing an immediate context switch if needed.

## NOT ALL REAL TIME OPERATING SYSTEMS ARE CREATED EQUAL!

CMX-RTX is a powerful RTOS that uniquely provides:

- ◆ The smallest footprint
- ◆ The fastest context switch times
- ◆ The lowest interrupt latency times
- ◆ True preemption
- ◆ Scheduler and interrupt handler written in assembly for speed and optimization
- ◆ Optional co-operative and time-slicing scheduling
- ◆ Nested interrupts
- ◆ All functions contained in a library
- ◆ Interrupt callable functions
- ◆ Scalability
- ◆ Several C vendors supported

## NOT ALL COMPANIES ARE CREATED EQUAL

CMX technical support is renowned throughout the world. Average wait times for a tech support person is under one minute and over 95% of our calls are resolved over the phone! If a question cannot be resolved over the phone, the answer is usually found and relayed to the customer within hours.

Our philosophy also includes giving the engineer all of the source code to the product. This is not only an invaluable debugging tool, but dramatically reduces the learning curve associated with any RTOS. It also allows for smaller code size because only functions that are used are linked into the final output module.

## A PARTIAL LISTING OF CMX-RTX FUNCTIONS

### TASK MANAGEMENT

Create a task.  
Remove a task.  
Start a task.  
Suspend a task, with time-out provision.  
Wake a suspended task.  
Forcefully wake a task.  
Change a task's priority.  
Terminate a task early.  
Do a cooperative rescheduling.  
Disable task scheduling.  
Enable task scheduling.

### EVENT MANAGEMENT

Wait on event(s), with time-out provision.  
Set an event.  
Clear an event.

### SEMAPHORE MANAGEMENT

Get semaphore.  
Pend for semaphore, with time out provision.  
Post to semaphore.  
Flush semaphore.

### MESSAGE MANAGEMENT

Get a message.  
Wait for a message, with time out provision.  
Send a message.  
Send a message, wait for reply.  
Wake task that sent message, if waiting on reply.  
Wait on Mailbox(s), with time out provision.

### QUEUE MANAGEMENT

Create a circular queue.  
Reset queue to empty.  
Add to top of queue.  
Add to bottom of queue.  
Remove from top of queue.  
Remove from bottom of queue.

### RESOURCE MANAGEMENT

Get a resource.  
Reserve a resource, with time out provision.  
Release a resource.

Automatic priority inversion.

### MEMORY MANAGEMENT

Create a fixed block pool.  
Request free block from pool.  
Release block back to pool.

### TIMER MANAGEMENT

Create a cyclic timer.  
Change a cyclic timer event parameters.  
Start a cyclic timer.  
Restart a cyclic timer.  
Stop a cyclic timer.  
Restart a cyclic timer with new initial time period.  
Restart a cyclic timer with new cyclic time period.

### SYSTEM MANAGEMENT

Initialize CMX.  
Enter CMX.  
Enter interrupt.  
Exit interrupt.  
Enter power down mode.

**Contact CMX for a current list of supported processors.**

### *A Few Customers of CMX...*

- ◆ AMD
- ◆ IBM
- ◆ Sony
- ◆ Baxter
- ◆ Philips
- ◆ TV/COM
- ◆ Analog Devices
- ◆ Fujitsu Telecom
- ◆ Ericsson Mobile
- ◆ Nokia Telecomm
- ◆ Invensys

- ◆ ITT
- ◆ AMP
- ◆ Ford
- ◆ Boeing
- ◆ U.S. Navy
- ◆ U.S. Robotics
- ◆ AT&T Wireless
- ◆ Temic Telefunken
- ◆ Hewlett Packard
- ◆ Emerson Appliance
- ◆ Bose Corporation

- ◆ TRW
- ◆ Enraf
- ◆ Xerox
- ◆ Siemens
- ◆ Rockwell
- ◆ Kenwood
- ◆ Honeywell
- ◆ ABB Power
- ◆ Benefon OY
- ◆ Allied Signal
- ◆ Hughes Network

## WHAT IS CMX TCP/IP?

CMX TCP/IP is a portable, high performance TCP/IP implementation for embedded systems. Memory usage is localized and deterministic. CMX TCP/IP uses RTOS signaling mechanisms to provide a true, multi-tasking re-entrant network stack. When no RTOS is available, CMX TCP/IP provides a straight forward, single threaded stack that supports multiple sockets via the select multiplexer. The system has a simple and familiar programming API, including standard sockets and callable application modules.

CMX TCP/IP is the only solution that you will need for embedded applications. It is embedded in set top boxes to connect thousands of homes to the internet through their cable TV provider. It links thousands of sites to a home office for interactive, satellite-based training. It connects instruments to a PC or workstation over ethernet. It links gaming tables and industrial controllers on LAN's to host computers.

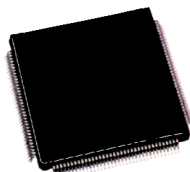
CMX TCP/IP is sophisticated enough to handle the toughest wide area networking jobs, but is still compact and economical enough to be used in a simple LAN application. CMX TCP/IP's low cost, clean implementation, and high functionality make it a breeze to add TCP/IP networking to any application.

**CMX TCP/IP**  
**is a 100% RFC**  
**compliant**  
**stack**

### Complete Solution for Embedded Processors

#### CMX-RTX

**Real Time Multi-Tasking Operating System**



**Many 8/16/32 Bit Microcomputers, Microprocessors and DSPs Supported**

#### CMX TCP/IP

##### Client/Server

FTP, Telnet, Web, SNMP, Mail

##### Socket Interface

BSD Compatible Sockets

##### Protocols

TCP, IP, UDP, ICMP, ARP, IGMP

##### Data Link Layer

Ethernet, Slip, PPP

##### Physical Layer

Drivers

## WHY USE CMX TCP/IP?

When selecting networking tools for your embedded project, there are technical decisions to be made and there are business decisions to be made. Once you have determined that the tool is technically appropriate, it is time to consider the following questions:

- ◆ Is full source code provided with the system?
- ◆ Are there no royalty payments for deployed products?
- ◆ Can the system be run without purchasing an RTOS, if desired?
- ◆ Is the software fully documented and supported?
- ◆ Is training readily available?
- ◆ Is the system economically priced?

With CMX TCP/IP, you can happily answer YES to all of the above questions. CMX is in the business of providing tools that not only make sense to engineers but to management, as well!

## WHY CMX?

CMX technical support is renowned throughout the world. Average wait times for a tech support person is under one minute and over 95% of our calls are resolved over the phone! If a question cannot be resolved over the phone, the answer is usually found and relayed to the customer within hours.

Our philosophy also includes giving the engineer all of the source code to the product. This is not only an invaluable debugging tool, but dramatically reduces the learning curve associated with any TCP/IP stack.

## CMX TCP/IP INCLUDES:

- ◆ TCP
- ◆ UDP
- ◆ IP
- ◆ ICMP
- ◆ ARP
- ◆ IGMP
- ◆ SLIP
- ◆ ETHERNET
- ◆ Standard BSD Socket Interface



## CMX TCP/IP Add On Options

♦ DHCP Client	Dynamic Host Configuration Protocol Client
♦ DHCP Server	Dynamic Host Configuration Protocol Server
♦ FFS	Flash File System
♦ FTP	File Transfer Protocol Client/Server
♦ IMAP4	Internet Messaging Access Protocol
♦ NAT	Network Address Translation
♦ POP3 Client	Post Office Protocol
♦ PPP	Point to Point / Serial Link Internet Protocols
♦ PPPoE	PPP Over Ethernet
♦ SMTP	Simple Mail Transfer Protocol
♦ SNMP V2	Simple Network Management Protocol V2 Agent (includes V1)
♦ TELNET Client	Telnet Client (includes Telnet Server)
♦ TELNET Server	Telnet Server
♦ TFTP	Trivial File Transfer Protocol Client/Server
♦ WEB Client	Embedded Web Client (includes WEB Server)
♦ WEB Server	Embedded Web Server (HTTP Server)

CMX TCP/IP offers Ethernet drivers for the most popular ethernet controllers available for the embedded industry. The library of drivers also includes drivers for supported target processors that offer an onboard MAC with the chip. Please contact CMX for the latest list of Ethernet Drivers and/or Wireless Ethernet functionality for CMX TCP/IP. For processors that have built-in ethernet capabilities, the drivers have been specifically developed and are also provided with CMX TCP/IP.

**Contact CMX for a current list of supported processors.**

## IMPORTANT FEATURES

- |  |                          |
|--|--------------------------|
| ♦ Free Source Code                                 | ♦ Easy to Use            |
| ♦ No Runtime Royalties                             | ♦ No RTOS Required       |
| ♦ Unlimited Users per Site                         | ♦ Full Documentation     |
| ♦ Full Featured Regression Test and Sample Program | ♦ Free Technical Support |
|  | ♦ Economical             |

## CMX-Tiny+ RTOS Minimizes RAM Usage for 8-, 16, 32-bit processors!

The CMX-Tiny+ real time multi-tasking operating system is an extremely "lean and mean" kernel that provides an optimized, small footprint solution for 8-, 16-, 32-bit processors with limited RAM. This specially designed RTOS allows the user to develop application code that is run under an RTOS and yet only use the onboard RAM that the processor provides! CMX-Tiny+ does not need any external RAM, regardless of whether the processor can support the use of external RAM or not.

CMX-Tiny+'s code size is so small that it allows the processor's onboard FLASH to support both the user's application code and the CMX-Tiny+ code, in most cases. This unique RTOS, based on a scaled down version of the popular CMX-RTX™, retains most of the power of CMX-RTX as well as the more frequently used functions. CMX-Tiny+, a truly preemptive RTOS, also provides support for cooperative scheduling, if desired. CMX-Tiny+ also is integrated with the CMX-MicroNet™ TCP/IP stack for those applications requiring networking connectivity.

### CMX-Tiny+ Features

- ◆ **Extremely Small FLASH/RAM Footprint**
- ◆ **Truly Preemptive RTOS**
- ◆ **Low Power mode supported**
- ◆ **Full Source Code With Every Purchase**
- ◆ **Free Technical Support and Updates**
- ◆ **Low, Economical Pricing**
- ◆ **No Royalties on Shipped Products**
- ◆ **Integrated with CMX-MicroNet for Optional Networking Connectivity**

**Contact CMX for a current list of supported processors.**

## A Partial Listing of CMX-Tiny+ Functionality

- **Task Management**
- **Message Management**
- **System Management**
- **Event Management**
- **Resource Management**
- **Timer Management**





# CMX Add-In Modules

## CMXKAware™ Kernel Awareness

Embedded systems engineers have long desired the ability to view their programming applications running under an RTOS via their In-Circuit Emulators and Simulators/ROM monitor debuggers. To meet this important need, CMX has developed CMXKAware which integrates seamlessly with a number of the leading emulator manufacturers' and/or C vendors' simulators/ROM monitor debuggers. The result is a dramatically enhanced debugging capability that will help to minimize application development time, thereby reducing time to market for companies that manufacture products which include embedded systems.

CMXKAware is an Active X object, DLL or ORTI (OSEK Run Time Interface) that presents all of the RTOS-specific information on the screen. CMXKAware allows you to display CMX-RTX's internal data structures in a convenient series of lists in the RTOS window of the debugger. This provides you with information about each of the active tasks in the target application, about each semaphore, resources, mailbox, queue and event flag group along with a list of all the tasks waiting on these kernel objects, and more. Data can be displayed graphically in real-time.

The new OSEK Run Time Interface (ORTI) for CMXKAware is a universal interface for development tools to the CMX-RTX RTOS. This powerful and flexible interface allows for the evaluation and display of information about CMX-RTX, its state, its performance, the different task states, the different operating system objects etc. The object information is provided via an ASCII text file and since these implementations are configured statically, this data will be available at compilation. Additionally, the ORTI file contains dynamic information as a set of attributes that are represented by formulas to access corresponding dynamic values. Formulas for dynamic data access are comprised of constants, operations, and symbolic names within the target file. By performing a continuous scan of the internal data structures described in the ORTI file, debugging tools can extract and display critical kernel-relative information, such as task states and event traces for the last four RTOS calls in the application under test. Further, the CMXKAware ORTI Builder (see graphic below) automates the creation of ORTI files adapted to the CMX-RTX RTOS for debugging purposes, thereby allowing embedded engineers to focus their efforts exclusively on debugging their application.

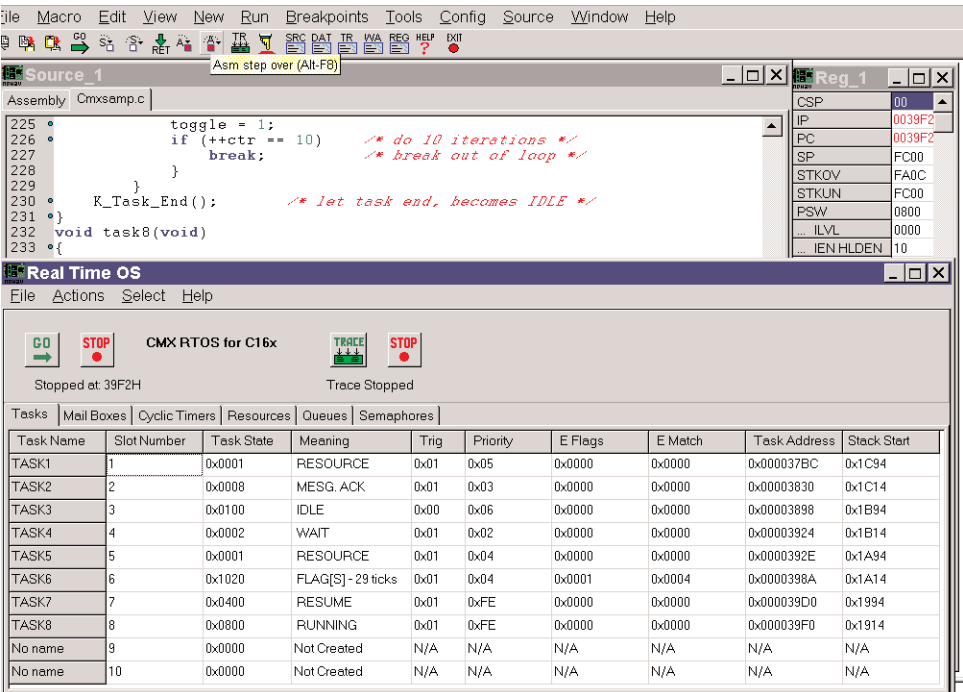
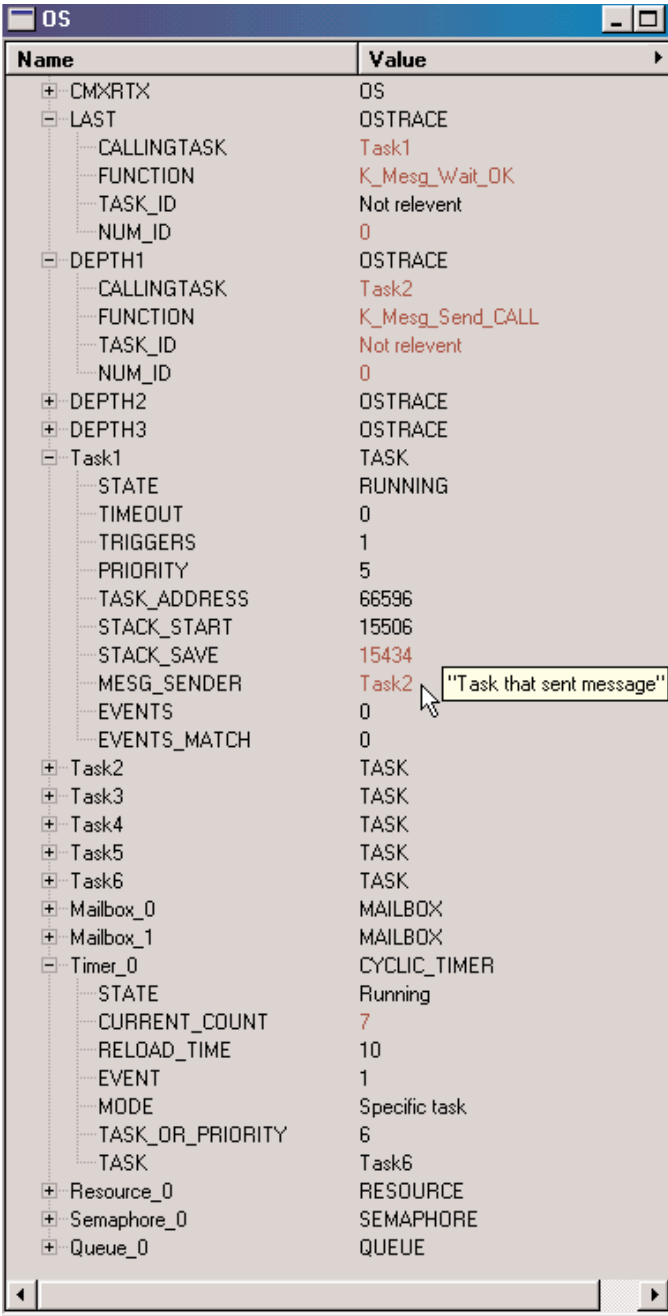
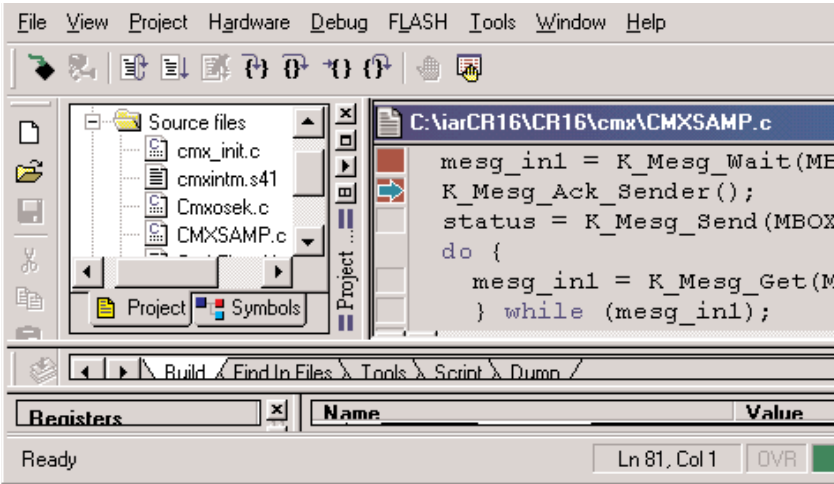
Since the OCX, DLL or ORTI utilizes the emulator or simulator kernel aware API, no target resources are used, as in the case with CMXBug (shown on another page). Most emulators and simulators/ROM monitor debuggers are supported. Please contact CMX for a complete list of supported platforms.

## CMXKAware ORTI Builder

Task Name	Task Slot number
Task1	task1_slot
Task2	task2_slot
Task3	task3_slot
Task4	task4_slot
Task5	task5_slot
Task6	task6_slot

# Emulator displays with CMXKAware

The screen shots below and right illustrate an emulator displaying an OSEK ORTI file generated by the CMXKAware ORTI builder. As you can see, ALL aspects of the RTOS can be seen and viewed. Also, if enabled by the user, CMXKAware offers the ability to display the last four RTOS calls, which can be a powerful 'trace' capability for debugging.



The screen shot on left illustrates another emulator displaying RTOS functionality via an Active X file. As shown in the graphic, ALL aspects of the RTOS can be seen and viewed.



# CMX Embedded Storage

## CMX-Embedded Storage Products

CMX has produced a range of products tailored to satisfy the needs of very different embedded products.

There are a wide variety of target storage media types and flash devices for embedded systems, many with different characteristics. The target embedded devices also have widely varying available resources. CMX has produced a range of file system products get the best out of these environments:

**CMX-FFS** - failsafe flash file system for embedded systems supporting NOR, NAND and DataFlash devices.

**CMX-FFS-TINY** - failsafe flash file system for micros with limited resources supporting small erasable flash.

**CMX-FFS-FAT** - FAT12/16/32 compatible file system supporting standard media.

**CMX-FFS-THIN** - FAT12/16/32 compatible file system for micros with limited resources supporting standard media.

**File Systems Summary Table**

	Fail-safe	FAT Compatible	RAM Usage	ROM Usage	Typical MCUs	Storage Media
CMX-FFS	Y	N	(1)	30-35K	16/32bit	NAND Flash NOR Flash Serial Flash DataFlash RAM
CMX-FFS-TINY(2)	Y	N	<<1K	6-12K	8/16bit	Serial Flash Small erasable sector flash RAM
CMX-FFS-FAT	N	Y	2.5K-??	15-30K	16/32 bit	Compact Flash Cards MMC/SD cards HDD's NAND flash (with CMX-FFS-FAT-(S)FTL) DataFlash (with CMX-FFS-FAT-DF(M))
CMX-FFS-THIN	N	Y	<700B - 4K	4-15K	8/16 bit	Compact Flash Cards MMC/SD cards NAND flash (with CMX-FFS-THIN-(S)FTL) DataFlash (with CMX-FFS-THIN-DF(M))

(1) Dependent on flash type - typically between 20K and 100K

(2) See section below for various types

# Flash File Systems

**CMX-FFS** - Is the standard edition flash file system for embedded systems developers and includes 100% Failsafe, Multiple volumes, Wear-leveling, Directories, Boot sector support, and RAM Drive. Additional drivers for CMX-FFS include NOR Flash Driver, NAND Flash Driver and Atmel DataFlash Driver.

**CMX-FFS-TINY** is specifically designed for certain characteristics in the flash being used. In this way it is possible to get efficient, fail-safe flash on devices with minimal resources.

**CMX-FFS-TINY-BW** This system is for use with any flash devices with small erasable sectors (typically <4K) and in which data can be written in either byte or word units. Typical devices include MSP430 internal flash, SST serial flash and many more.

**CMX-FFS-TINY-ST** This system is for use with ST serial flash for data.

**CMX-FFS-TINY-DF** This system is for use with Atmel DataFlash.

**CMX-FFS-FAT** - Is a full-featured file system for embedded systems developers who wish to add devices to their products that require FAT12/16/32 compliant media to be attached to them and includes Long filenames, Multiple Volumes, and the following Drivers: Compact Flash (True IDE and Memory I/O), MMC/SD (SPI S/W or H/W) and RAM. Additional drivers for CMX-FFS-FAT include Flash Translation Layer for NAND Devices, and a DataFlash Management Layer for Atmel DataFlash and for multiple Atmel DataFlash devices.

**CMX-FFS-THIN** - Is a scalable, reduced footprint file system designed for integration with 8-bit or 16-bit embedded processors with limited resources and includes FAT 12/16/32, Long file names, and the following Drivers: Compact Flash (True IDE and Memory I/O), MMC/SD (SPI S/W or H/W) and RAM. Additional drivers for CMX-FFS-THIN include Flash Translation Layer for NAND Devices, and a DataFlash Management Layer for Atmel DataFlash and for multiple Atmel DataFlash devices.

## Flash Management Software

### CMX-FTL

This is a fully featured Flash Translation Layer which allows NAND flash to interface directly to a FAT file system. Typically it requires 3-20K bytes of RAM depending on flash type and configuration.

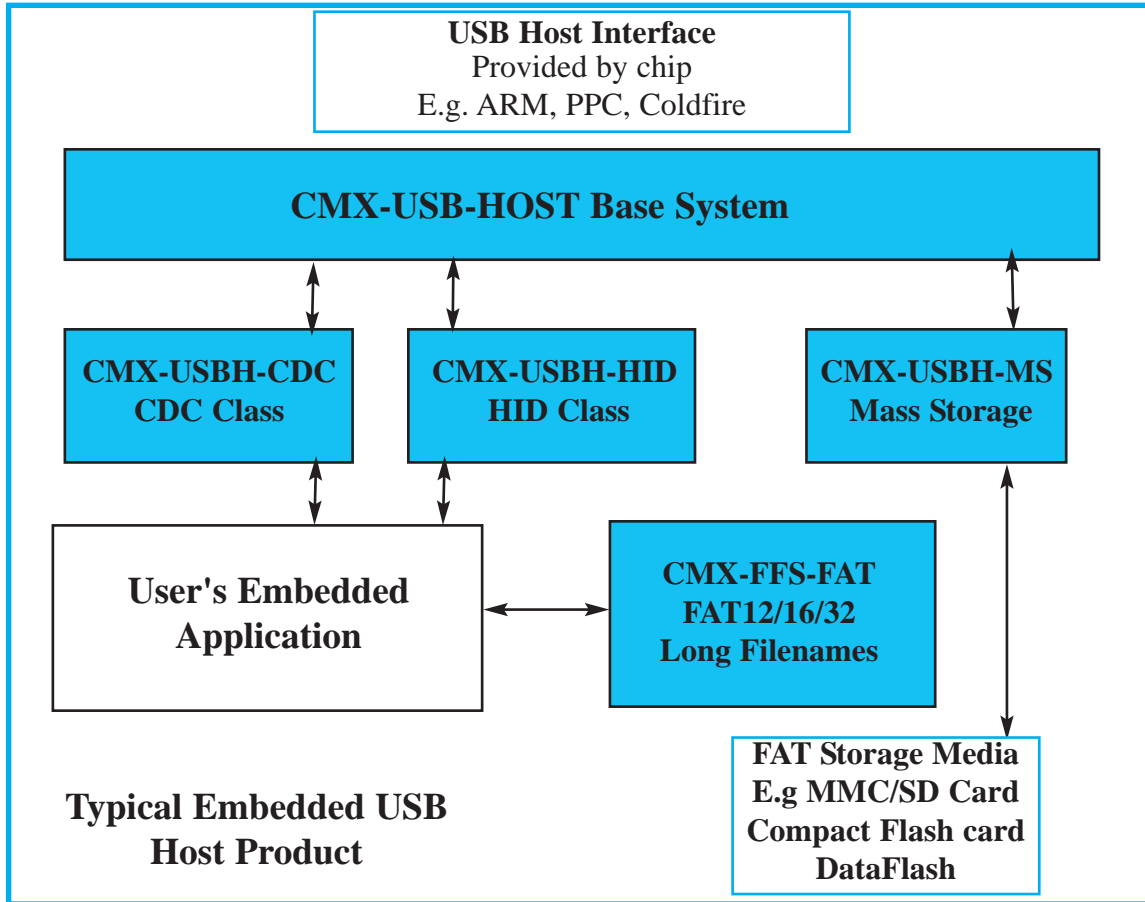
This may be ordered standalone, as driver for CMX-FFS-FAT (CMX-FFS-FAT-FTL) or as a driver for CMX-FFS-THIN (CMX-FFS-THIN-FTL).

### CMX-DFML

All of CMX's file systems can be ordered with our unique DataFlash Management Layer which gives added reliability to storage on Atmel DataFlash devices. Single or Multiple DataFlash (arranged in a single flash array) are capable.

## CMX-USB Host - Overview

CMX Systems is a well-known provider of Embedded Software Solutions. CMX-USB-Host is targeted at systems which need USB Host connectivity and the USB interface is provided to the End Point management level. CMX-USB Host Software provides an integrated suite of USB Host functionality that also includes pre-built packages for several targets. Adding USB Host connectivity just got easier for the embedded developer.



## CMX-USB-Host Base System

CMX-USB-Host (OHCI or EHCI or Non-Standard) is designed both for microcontrollers with integrated USB host controllers and for external USB host controllers. Designed as a small footprint, efficient driver - it works seamlessly with any host controller. The CMX-USB-Host package is fully compatible with CMX's Host Class drivers for Mass Storage, CDC, HID and PrinterLite. CMX-USB-Host Mass Storage is fully compatible with CMX-FFS-FAT.

## **CMX-USBH-CDC**

The CMX-USBH-CDC Class enables the designer to provide a virtual USB serial port on the embedded target which acts as a standard serial port and can be accessed using simple, serial API functions. CMX-USBH-CDC is optimized to use minimal RAM and ROM and requires just the control channel endpoint and a single interrupt endpoint.

## **CMX-USBH-HID**

The CMX-USBH-HID Class enables the designer to provide support for Human Interface Devices such as a mouse, keyboard or joystick.

## **CMX-USBH-MS Mass Storage**

This allows any flash drive to plug into the host system and to be read by the hosts file system. The system allows the host to access the drive using the minimum possible RAM and ROM requirements. It can be used with both an integrated USB host controller or with an external USB host controller. CMX-USBH-MS uses just the control channel endpoint and two bulk endpoints. CMX-USBH-MS is fully compatible with CMX-FFS-FAT Flash File System product.

## **CMX-USBH-PL**

The CMX-USBH-PL Class enables the designer to provide support for Printers. This is a Printer Lite class.

## **File Systems**

CMX also provides file system solutions for embedded systems. In particular the CMX-FFS-FAT product is a optimal solution for running fully featured FAT file systems on Microcontrollers. The standard features include FAT12/16/32 and long filenames which make it as simple as possible to access standard flash devices both from your embedded application and through USB mass storage.

## **Integrated Packages**

CMX offers complete pre-ported packages for several Microcontrollers with integrated USB HOST and external USB Host controllers. A complete tested project can be delivered including reference schematics which means that the developer gets an instant success in the project and can immediately focus on their core competences.

Please contact CMX for a current list of devices.

These packages work on various standard evaluation boards. Please contact CMX for more information about this.

## **CMX Systems, Inc.**

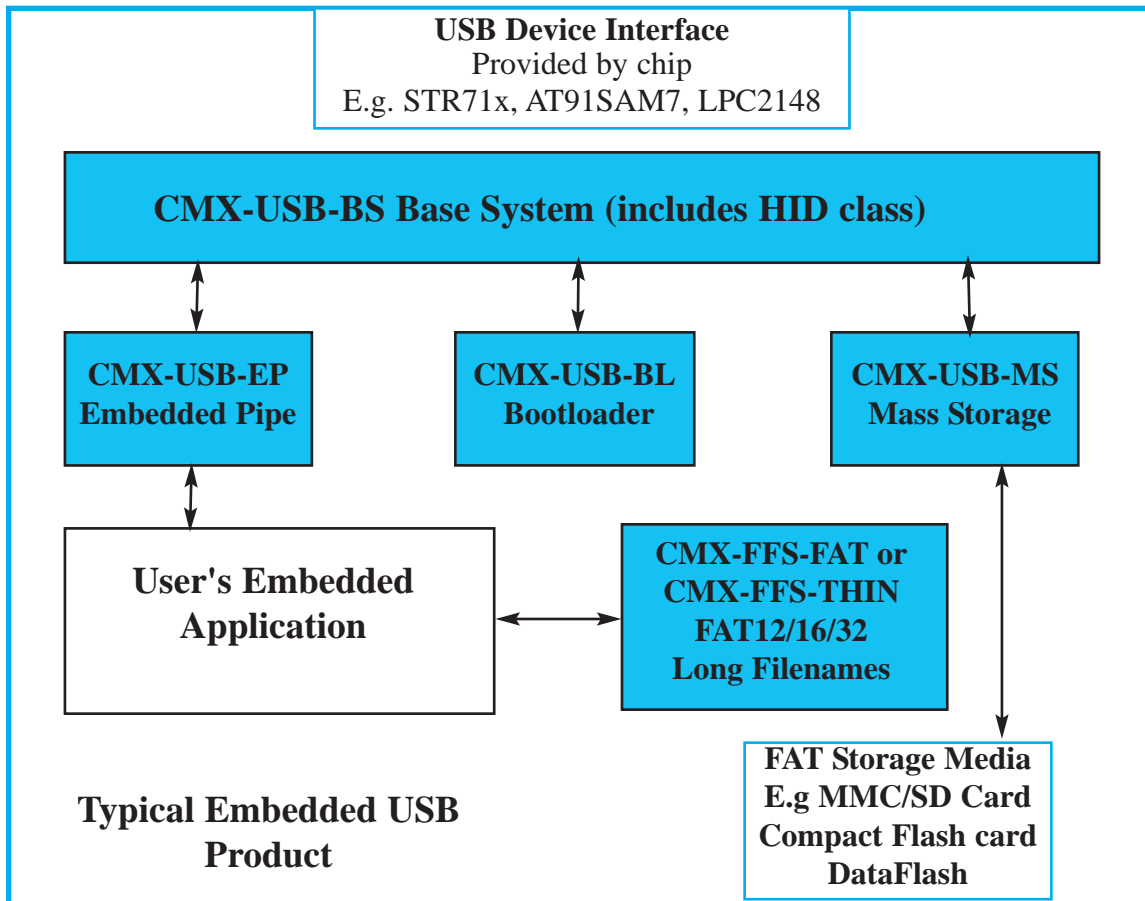
CMX Embedded Software Products are shipped royalty free, with full 'C' source code, 6 months technical support and software updates.

**Contact CMX for a current list of supported processors.**



## CMX-USB - Overview

CMX Systems is a well-known provider of Embedded Software Solutions. CMX-USB is targeted at systems which need USB device connectivity and the USB interface is provided to the End Point management level. CMX-USB provides an integrated suite of USB device functionality that also includes pre-built packages for several targets. Adding device USB connectivity just got easier for the embedded developer.



## CMX-USB Base System

CMX-USB-BS handles all the basic setup and management of the USB system. The system assumes a USB Device Controller which handles all of USB up to the end point level. Also, part of the CMX-USB-BS system is the USB configuration including: Vendor ID, Product ID, End Point Type, End Point Addresses, Bulk, Interrupt, Control channels.

## CMX-USB-MS Mass Storage

CMX-USB-MS storage allows you to connect standard PC compatible storage media attached to your embedded system like a standard pen-drive. The package includes the USB SCSI layer and also the low level drivers for attaching media to the device. Possible media you can attach include: Compact Flash cards, MMC/SD cards, HDDs, DataFlash and NAND flash. CMX provides tested drivers and reference schematics for all these media types.

## **CMX-USB-CDC Class**

CMX-USB-CDC provides a CDC class implementation for supporting standard comm ports. An API is provided which allows the comm port to be either virtual in your embedded application or connect it to a serial port on your embedded design.

Provided is a sample application, which has an embedded command line terminal which can be accessed by the standard Windows terminal program (e.g. HyperTerm) through a Comm port over USB interface.

## **CMX-USB Bootloader**

CMX-USB-BL is the reliable way to allow field upgrades of your product.

The optional CMX-USB-BL code is a standalone system which allows an application to download new application code to the target. The footprint of CMX-USB-BL is less than 8K bytes and so can be held as a permanent, reliable boot-stub on a single small flash sector. This can then handle all future firmware upgrades.

## **File Systems**

CMX also provides file system solutions for embedded systems. In particular the CMX-FFS-FAT and CMX-FFS-THIN products are optimal solutions for running fully featured FAT file systems on Microcontrollers. The standard features include FAT12/16/32 and long filenames which make it as simple as possible to access standard flash devices both from your embedded application and through USB mass storage.

## **Integrated Packages**

CMX offers complete pre-ported packages for several Microcontrollers. A complete tested project can be delivered including reference schematics which means that the developer gets an instant success in the project and can immediately focus on their core competences.

These packages work on various standard evaluation boards. Please contact CMX for more information about this.

## **CMX Systems, Inc.**

CMX Embedded Software Products are shipped royalty free, with full 'C' source code, 6 months technical support and software updates.

**Contact CMX for a current list of supported processors.**

# Bootloaders

## CMX-BL-FAT

The CMX-BL-FAT can be used as a standalone Reliable Bootloader for any sector based storage media such as SD / MMC cards. This Bootloader reads the configuration file and writes to the target.

CMX-BL-FAT also provides a bootloader, when using media such as a pen-drive, along with the CMX-USB-Host and CMX-USBH-MS.

# Window Drivers

## CMX-USB-EP Embedded USB Pipe

CMX-USB-EP: The easy way to connect your embedded application to your host application!

The CMX-USB-EP module is designed to allow your embedded application to communicate with your host application without needing to be concerned about USB. On the embedded side, the application simply opens, reads and writes to a pipe. CMX also provides a Windows driver which creates standard Comm Objects on the PC which any application or script can simply read or write. Sample applications are provided for a number of languages including VC++ and VBScript.

## Integrated Packages

CMX offers complete pre-ported packages for several Microcontrollers. A complete tested project can be delivered including reference schematics which means that the developer gets an instant success in the project and can immediately focus on their core competences.

These packages work on various standard evaluation boards. Please contact CMX for more information about this.

## CMX Systems, Inc.

CMX Embedded Software Products are shipped royalty free, with full 'C' source code, 6 months technical support and software updates.

**Contact CMX for a current list of supported processors.**



## Integrated or Stand Alone CANopen Functionality from CMX!

CMX-CANopen implements the CANopen protocol stack as defined by the CiA (CAN in Automation manufacturer's group) Draft Standard "CANopen Application Layer and Communication Profile" DS301 version 4.02. Although primarily intended for the usage in CANopen slaves, the code can also be used to implement a CANopen NMT Master (Network Management Master). Code examples provided have been tested with and passed the official CANopen conformance test.

CMX-CANopen can be used both as a stand-alone product as well as a communication stack for the real-time operating system CMX-RTX. When used with CMX-RTX task priorities can be adapted to best suit the needs of a particular application allowing to assign different priority levels to different process data messages.

Many define statements are used to enable and disable CANopen features, allowing for a very high-level of customization. Single transmission types can be selected to only include those transmission types that are actually used in the application.

### Functionality Provided

- ◆ **Object Dictionary and SDO Functionality**
  - One SDO server
  - Expedited SDO transfer
  - Segmented SDO transfer
- ◆ **NMT Services**
  - Full NMT slave state machine
  - Node Guarding client
  - Heartbeat producer
  - Heartbeat consumer
- ◆ **PDO Functionality**
  - Event timer
  - Change-of-state with inhibit time
  - Synchronized transfer
  - Dynamic or static PDO communication parameters
  - Dynamic or static PDO mapping parameters
  - Maximum of 1024 PDOs supported
- ◆ **Example Implementations Provided**

### Features

- ◆ **CANopen Conformance Test Compliant**
- ◆ **Can Be Run Stand Alone or with CMX RTX**
- ◆ **Complete Documentation with Examples**
- ◆ **Full Source Code With Every Purchase**
- ◆ **Free Technical Support and Updates**
- ◆ **Low, One-time License Fee**
- ◆ **No Royalties on Shipped Products**



# About CMX

---

## Background

Since its inception in 1990, CMX Systems, Inc. (formerly known as CMX Company) has focused on providing its customers with all of the tools needed to program their embedded applications. These applications range from automotive, medical equipment, consumer electronics, communications, to aerospace, manufacturing automation, and many other industries. The company's business is to develop and support real-time multi-tasking operating systems (OS's) for a wide variety of 8-, 16-, 32- and 64-bit microprocessors, DSP's and microcomputers. CMX enhances its RTOS with an optional TCP/IP networking package and additional networking add-on components.

CMX also develops and supports products that enhance the user's ability to create, test and debug application code. In addition to developing OS's and tools, CMX also distributes a variety of C tools including compilers, assemblers, linkers, librarians, simulators and ROM debuggers. The company's CMX-RTX Real-Time Multi-Tasking Operating System supports more than 40 processor families and more than 30 C-compiler vendors.

### A Message from the President

*Thank you for considering CMX real time software. Our company's sole mission is to develop and meticulously support the world's best real time software products. We take great pride in participating in the ultimate success of our clients who are constantly finding new and innovative uses for our software.*

*On behalf of the entire staff of CMX, I would like to welcome you to join our family of satisfied users. I promise you that we will never lose our focus on your complete satisfaction.*

- Chuck Behrmann

## Our Commitment to You!

### Service and Support

CMX is dedicated to providing our customers with personal attention, quality, commitment, and meticulous support in the use of our products. Because of this philosophy, CMX technical support is renowned throughout the world for its thoroughness and fast resolution of problems. Our quality control statistics indicate that over 95% of technical support calls are resolved over the phone on the first call. In the rare case in which the question cannot be answered immediately on the phone, the correct answer is usually found and relayed to the customer within hours.

We also strongly believe that the source code for the software be given to our customers with every purchase. Not only does this afford the user with an invaluable debugging tool, it also greatly reduces the learning curve normally associated with any real time software product.

### Training

While many of our users are able to begin using our software after spending a short amount of time with our examples and documentation, some companies prefer to have formal training for their staff.

CMX has made a special effort to employ expert trainers with years of hands-on experience in the embedded systems industry. We offer standard training and/or customized training courses to suit the needs of the smallest design firm or the largest multi-national engineering corporation.

Call us for more information about our currently available training courses.