68K/ColdFire

SOFTWARE DEVELOPMENT TOOLSET

TASKING™

HIGHLIGHTS

- **Supports 68K/CPU32 and ColdFire**
- **Fully-integrated embedded development environment**
- **Easy project setup and management**
- **Language sensitive editor**
- **Highly optimizing C/C++/EC++ compilers**
- **CrossView Pro debugger**
  - **Instruction set simulator**
  - **ROM monitor**
  - **Background Debug Mode**
  - **(OSEK) kernel aware**
- **Available for:**
  - **PC/Windows**
  - **Sun/Solaris**

THE TASKING 68K/COLDFIRE SOFTWARE DEVELOPMENT TOOLSET

Altium is pleased to introduce the TASKING 68K/ColdFire Software Development Toolset. With over 25 years experience in cross-development tools, our TASKING toolsets are designed to meet the needs of embedded systems developers. By creating a total, integrated development environment, the 68K/ColdFire software development toolset can help you get your application development underway as rapidly and easily as possible so you can meet your time-to market deadline.

The 68K/ColdFire software development toolset consists of a C/C++/EC++ compiler, macro assembler, linker/locator, libraries, CrossView Pro debugger and EDE (Embedded Development Environment).

EMBEDDED DEVELOPMENT ENVIRONMENT

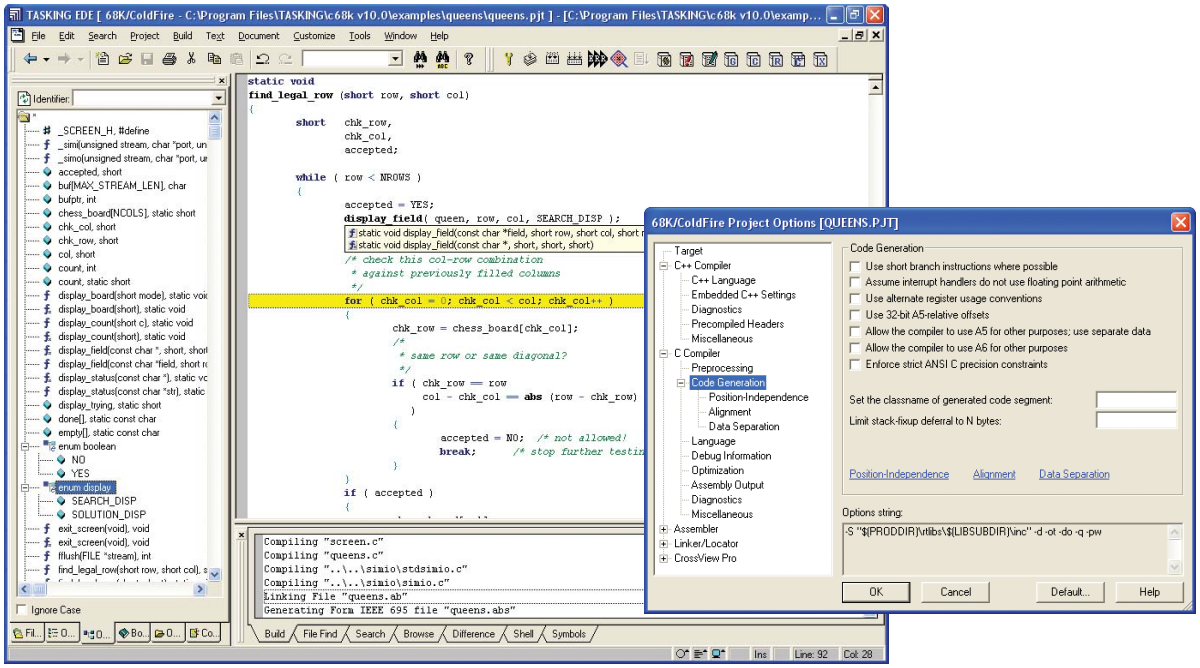
With the TASKING 68K/ColdFire EDE, you can create and maintain projects the easy way. All project related aspects, such as the application source files, the tool options (compiler, assembler, linker/locator, CrossView Pro debugger), file management and build process options, are managed from one central point. File dependencies as well as the sequence of operations required to build the application are handled automatically.

The EDE of the 68K/ColdFire toolset offers many productive features for application and code development, such as:

- **Project Spaces** enable you to group multiple projects in one view, thus improving project management of complex developments.
- **ChromaCoding** provides syntax coloring for programming languages such as C, C++ and assembly.
- **Tags Browsing** offers you a graphical overview of the applications' cross-references, and allows easy navigation through the available variables and functions.

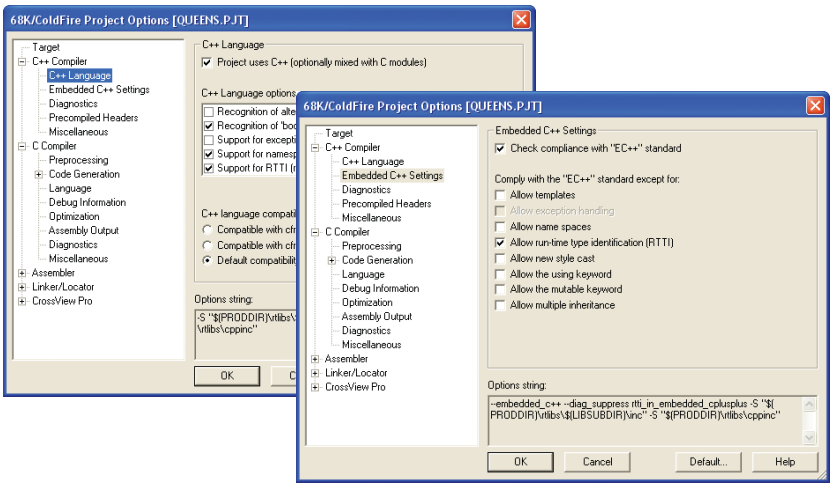
Supported 68K/ColdFire variants:

- **ColdFire series:**
  MCF5204 MCF5206 MCF5206E
  MCF5249 MCF5249L MCF5272
  MCF5280 MCF5282 MCF5307
- **DragonBall series:**
  MC68328 MC68EZ328
  MC68VZ328 MC68SZ328
- **CPU32 series:**
  MC68302 MC68306 MC68330
  MC68331 MC68332 MC68336
  MC68340 MC68360 MC68F375
  MC68376
- **68K series:**
  MC68000 MC68HC000
  MC68HC001 MC68EC000
  MC68SEC000 MC68008
  MC68010 MC68020
  MC68EC020 MC68030
  MC68EC030 MC68040
  MC68EC040 MC68LC040
  MC68V040 MC68060
  MC68EC060 MC68LC060



The TASKING EDE makes code development and project management easy

- **CodeFolio** allows you to easily insert 'snippets' of template code, thus increasing coding efficiency. It allows macro expansion and prompted input as you insert the code.
- **Selective Display** gives you a better overview of your source code. Allowing you, for example to collapse and expand functions at the braces level. Six formatting options are available.
- **HTML View Window** lets you browse the product manuals, project or code documentation or even surf the net without leaving the EDE.
- **Split Windows** provide full control over source code by allowing you to split your file horizontally or vertically into as many as four edit windows.
- **Right clicking** expedites a variety of tasks within the EDE (e.g., creating new files, adding files to a project, etc.).
- **CodeSense** advanced coding assistance offers rich type-ahead features, helping you in selecting the next expected function parameter or available structure members. When positioning your mouse pointer over a function name, the function prototype will be displayed.



Minimize overhead by using the scalability of EC++

C++/EC++ COMPILER

The C++ compiler delivers the power of object oriented design and coding to your 68K application. The object-oriented benefits of C++ can be incorporated into an existing C application, one module at a time, providing a smooth migration from C to C++. Full support of templates, dynamic casts, runtime type identification and exception handling is provided.

Scalable C++

Compatibility with the Embedded C++ (EC++) standard allows selective disabling of C++ features that may not be essential for your embedded application. By selecting (partial) compliance to the EC++ standard, code size overheads and runtime inefficiencies can be minimized.

General features of our C++/EC++ compiler include:

- **ANSI C and C++ compliant**
- **Embedded C++ (EC++) support**
- **Extensive optimizations**
- **In-line assembly**
- **Multipule locator output formats**

C Compiler

The C compiler applies powerful optimization algorithms which, combined with target-specific code generation, produce the most efficient code for your application. Complete symbol information can be generated for use with the CrossView Pro debugger, in-circuit emulators, or third party debuggers. The following libraries are also included: ANSI C, C++ library, runtime and software floating point (ANSI/IEEE 754-1985). The source code of all runtime libraries is provided so you can tailor the libraries to your specific needs.

Optimizations

The C/C++/EC++ compiler takes full advantage of the processor through a range of optimizations and code generation techniques. In addition, the compiler supports many advanced features which provide you with a high level of control for reducing the size of generated code and/or increasing execution speed.

The following are a sample of the optimization techniques used:

- **Processor-specific selection of addressing modes**
- **Automatic register variable allocation**
- **Common sub-expression elimination**
- **Strength reduction**
- **Invariant code motion**
- **Function in-lining**
- **Algebraic simplification**
- **Copy propagation**
- **Entry/Exit optimization**
- **Leaf function optimization**
- **Multiplication optimization**
- **Short branch instructions**
- **Subscript optimization**
- **Target Path computation**
- **Unreachable code elimination**
- **Dead store elimination**
- **In-line assembly language**
- **Position independence**
- **Processor-specific runtime libraries, including source**

Libraries

The C compiler tools contain all the necessary standard ANSI C and C++ libraries, runtime libraries, fixed and floating libraries. A fast floating point library is also included. Software floating point libraries support single and double precision (ANSI/IEE 754-1985) floating point operation. The single and double precision libraries are available without error trapping to optimize for speed.

ASSEMBLER AND LINKER/LOCATOR

Features include:

- **100% compatible with Motorola assembly**
- **Full macro facilities**
- **Supports structured and conditional assembly**
- **Unlimited number of relocatable, absolute, and combinab le segments**
- **Macro preprocessing**
- **Use of 8-bit, 16-bit or 32-bit addressing for forward branches**
- **Allows use of hardware floating point instructions**

The assembler translates assembly language source programs into object modules. The object modules can then be input into the linker/locator or catalogued in a library. The assembler has a built-in macro preprocessor which features an include file mechanism, macro definition and expansion.

Linker/Locator features include:

- **Automatic or user-defined allocation of code and data in memory**
- **Incremental overlaying techniques to reduce memory usage**
- **Locator overlaying**
- **Automatic inclusion of library modules**
- **Global type checking**
- **Generation of locator output in the following formats:**
  - **Motorola_CLAS COFF**
  - **Motorola S1, S2, S3 records**
  - **Intel HEX**
  - **Tektronix HEX**
  - **Extended Tektronix HEX**

Programming utilities

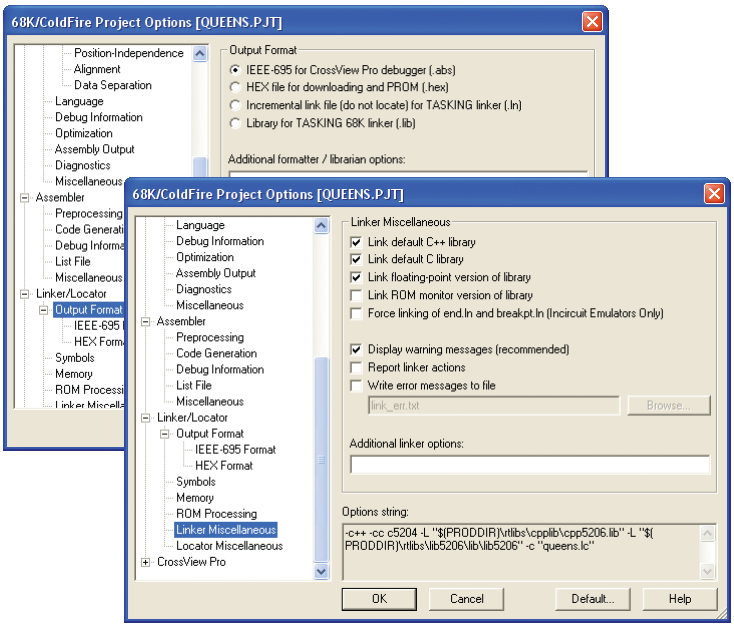
The C/C++/EC++ compiler tools also include the following programming support utilities:

- **Formatter – generates standard ASCII and binary download formats such as IEEE695, COFF, and S Records**
- **Librarian – create and maintain library files**
- **Symbol List Utility – lists debugging information on all types, global and local symbols with target locations for source lines of input code**
- **Global Symbol Mapper – displays addresses for all global symbols and segments within the current memory configuration**

Crossview Pro Debugger

In the world of embedded software development, there's no such thing as 'first time right'. Particularly with today's complex, high-pe formance applications, a fair share of development time will be spent within the debugger, validating your code.

The TASKING 68K/ColdFire CrossView Pro debugger is a perfect partner in checking, verifying and debugging your application. With its easy-to-use interface and powerful, extensive debugging features, CrossView Pro helps you debug your applications faster.



Controlling the linker and the output format

Features include:

- **Multi-window GUI**
- **C++/C/Assembly-level debugging**
- **Mixed mode source/assembly display**
- **Profiling/performance analysis**
- **Powerful breakpoint control**
- **Record and playback**
- **Multiple execution environments:**
  - **Simulator**
  - **BDM/JTAG**
  - **ROM monitor**

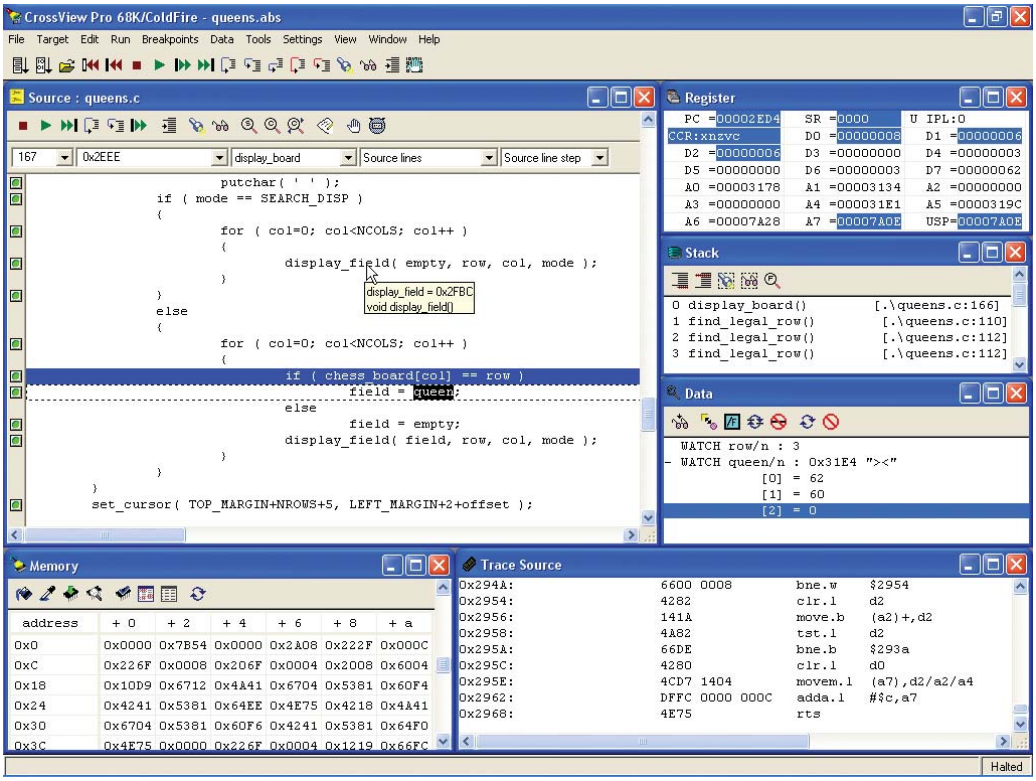
CrossView Pro provides multiple, resizable, and independently controlled windows. You choose the windows you need to view the different aspects of your code during debugging. It combines the flexibility of the C language with the control of code execution found in assembly language, which bringing functionality that reduces the time spent on testing and debugging.

Functionality includes:

- **Simple and advanced debugging features**
- **Intuitive source window**
- **Tracking scope and monitoring local variables**
- **Bubble-Spy™ for easy inspection of variables and functions**

Source window

The source window is the main debugging window. It allows you to view source; step through your application; set and clear breakpoints and assertions; watch and show variables; search for strings, functions, lines and addresses; call functions; evaluate expressions; and view performance analysis data. The source window can display code in C/C++ source, assembly or a mixed mode that allows a simultaneous view on your C/C++ source, intermixed with the corresponding assembly code. As to allow immediate access to your source files, you can jump directly from the CrossView Pro source window into the EDE editor at the exact source line.



Spend less time debugging with CrossView Pro

Multiple information windows

CrossView Pro offers a wealth of informative windows allowing you to navigate through your application, monitor and modify data objects, CPU registers, memory locations and the function-call stack.

The **data window** enables you to watch or show data, browse for locals or globals, double-click to modify values or to expand and contract complex data structures. Within this window, you can reformat (change display of radix and type) on an element-by-element basis. You can show or watch locals from any stack level, automatically track and display locals, and easily copy any variable as show or watch.

The **register windows** allow for the display and modification of CPU register values. Register windows are fully configurable to display any set of CPU registers. By defining multiple register windows you can easily organize your focus.

The **stack window** displays the contents of the function-call stack frame. You can easily configure stack-level breakpoints, navigate to the function call's source and monitor local variables for selected functions.

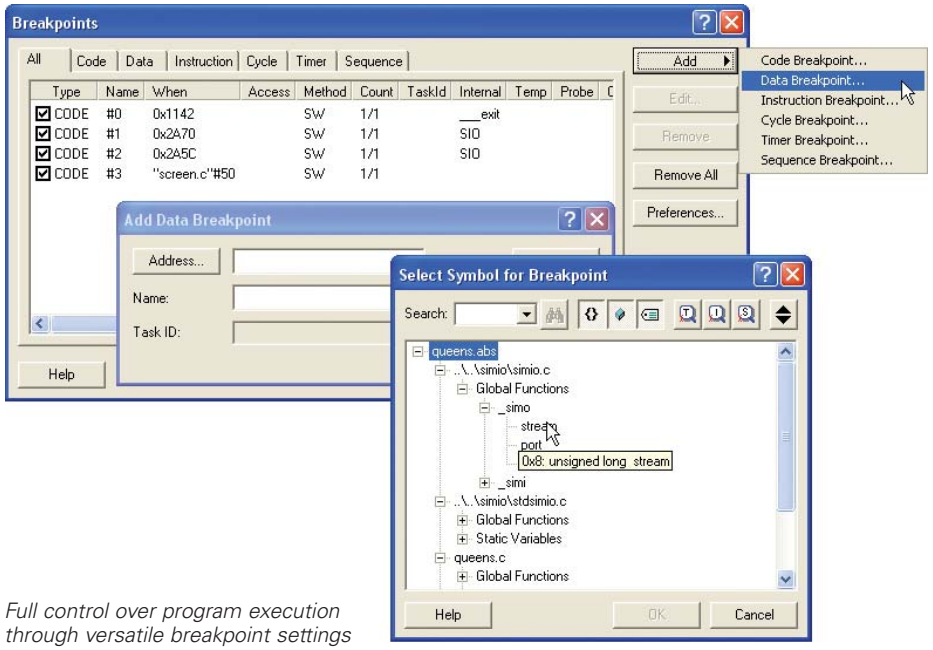
The **memory window** enables you to monitor and modify any memory location, providing complete control over the size and format of the data, as well as view coverage of the memory range.

Advanced breakpoints

Breakpoints halt program execution and return control to the user. In addition to industry standard code and data breakpoints, you can configure your application to halt based upon instruction counts, cycle counts, or timer counts. All types of breakpoints can be defined as 'stop-and-go' probe points.

Probe points briefly halt and immediately resume execution of the application. During the brief period that the application is halted, only user-specified actions will be performed. Through this mechanism, probe points allow least-intrusive debugging of time critical applications.

Finally, any number and type of breakpoints can be combined into so-called breakpoint-sequences. This allows easy specification of the most complex conditions that need examining.



Full control over program execution through versatile breakpoint settings

File system simulation

CrossView Pro I/O Simulation (IOS) allows for the use of standard ISO C system calls such as open(), read(), printf() and scanf() within your embedded application to interface with the host platform file I/O services.

When using IOS, you can read from and write to files on the host platform or a CrossView Pro Virtual I/O window directly. I/O Simulation will work in any CrossView Pro target execution environment.

Multiple execution environments

CrossView Pro supports three execution environments with the same standard interface.

- **Instruction set simulator**
  With the CrossView Pro instruction set simulator, you can debug your application on the host platform, even before your target hardware is available.
- **ROM monitor**
  The CrossView Pro ROM monitor debugger can be used with any commercial off-the-shelf evaluation board or customer target board. CrossView Pro, running on the host computer system, debugs your application on the target board through the monitor application via an RS232 interface.
- **Target debugging through BDM/JTAG**
  Taking advantage of the BDM facilities on the CPU32 and ColdFire derivatives, CrossView Pro offers high quality in-circuit-emulation functionality at a low cost. Via the host PC's parallel interface, CrossView Pro can communicate with the application directly. To connect the host to the application the Macraigor wiggler or P&E Microcomputer Systems interface cable can be used.

Program performance analysis

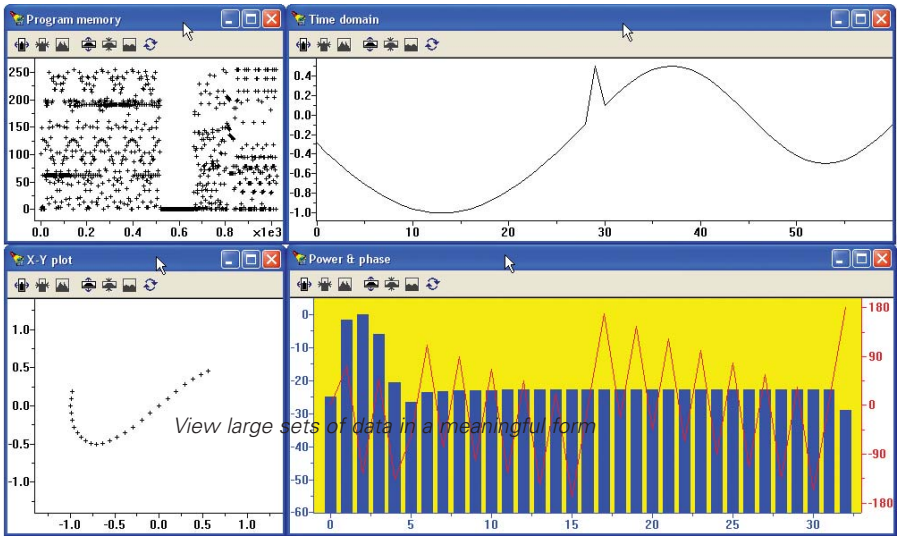
CrossView Pro provides a number of performance analysis capabilities to help you further optimize your application as well as shorten your debugging session.

- **Code coverage** enables you to check whether specific parts of your application code actually have been executed. Based on the code coverage reports you can build a complete test suite for your product and improve the quality of your application.
- **Profiling** allows you to perform timing analysis on the complete application or specific parts of it. Profiling information can be shown in the left margin of the source window, but can also be presented in a report, providing you with a full overview. Based upon the profiling information you can easily decide which functions should be optimized for speed.
- CrossView Pro's outstanding programmable **Graphical Data Analysis** simplifies quick detection of gross errors in signal processing routines, such as typically found in DSP applications. By displaying large sets of data in meaningful visual diagrams, CrossView Pro allows you to analyze the data without the need for reviewing or post-processing large files of raw data. You can also view the same set of data in several ways at the same time (e.g., in time- and frequency domain).

Four different analysis types are ready prepared: x-t plotting, x-y plotting, FFT (Fast Fourier Transformation) power spectrum and Eye diagram. The C-language scripts for these pre-defined graphs can be easily used as the basis for custom data analysis windows.

EASY DEBUGGING RTOS-BASED APPLICATIONS

Altium's Kernel-aware Debugging Interface (KDI) defines an open standard interface between CrossView Pro and an RTOS-Aware Debug Module (RADM). The RADM adds the capability to CrossView Pro to read, format and report kernel data structures. The KDI specification describes the open interface that can be used to add kernel-awareness to CrossView Pro for any commercial or proprietary RTOS. Check with the RTOS vendor of your choice for CrossView Pro RADM availability. The generic RADM for OSEK kernels included in the package is based on the ORTI 2.0 and 2.1 language specification.



View large sets of data in a meaningful form

CUSTOMER SUPPORT

When you purchase a TASKING product, it is the beginning of a long term relationship. Altium is dedicated to providing quality products and support worldwide. This support includes program quality control, product update service and support personnel to answer questions by telephone, fax or email.

A maintenance period is included with the purchase of TASKING products, which entitles you to enhancements and improvements as well as individual response to problems. Annual maintenance agreements are available to extend this initial support period.

AVAILABILITY

The 68K/ColdFire Software Development Toolset is available for PC/Windows and Sun/Solaris. Altium has a policy of continued improvement, for the latest information contact your local Altium Sales and Support Center.

PRODUCT PACKAGING & ORDERING CODES

Each TASKING product comes with full printed documentation. This documentation is also available on-line in the form of a Windows Help system, HTML and PDF and provides full-text search capabilities for quick and easy access to topics.

Product Code	Package Contents
07-200-001-012	EDE, C/C++/EC++ compiler, Assembler/Linker, CrossView Pro Simulator debugger
07-200-001-024	EDE, C/C++/EC++ compiler, Assembler/Linker, CrossView Pro BDM, ROM monitor and Simulator debugger (not on Sun/Solaris)
	Third party BDM interfaces:
07-290-001-001	Macraigor Wiggler for CPU32 5V
07-290-001-002	Macraigor Wiggler for CPU32 3V
07-292-001-003	P&E Micro BDM interface for ColdFire 5V
07-292-001-004	P&E Micro BDM interface for ColdFire 3V
07-292-001-005	P&E Micro BDM USB-MULTILINK interface for ColdFire 3V/5V

INTERNET

A trial version of the 68K/ColdFire Software Development Toolset is downloadable from our website at **www.tasking.com/68k**

ALTIUM SALES OFFICES

North America - Altium Inc

3207 Grey Hawk Court, Suite 100
Carlsbbad, CA 92010
Ph: +1 760-231-0760
Fax: +1 760-231-0761
Email: sales.na@altium.com

China – Altium Information Technology (Shanghai) Co., Ltd.

9C, East Hope Plaza
No. 1777 Century Avenue
Shanghai 200122
Ph: +86 21 6182 3900
Fax: +86 21 6876 4015
Email: sales.cn@altium.com

Australia – Altium Limited

3 Minna Close, Belrose
NSW 2085
Ph: +61 2 8622 8100
Fax: +61 2 8622 8140
Email: sales.au@altium.com

Germany – Altium Europe GmbH

Albert-Nestler-Straße 7
76131 Karlsruhe
Ph: +49 (0) 721 8244 300
Fax: +49 (0) 721 8244 320
Email: sales.de@altium.com

France – Protel AG

(A subsidiary of Altium Limited)

121 rue d'Aguesseau
92100 Boulogne-Billancourt
Ph: 0800 88 05 06
Fax: 0800 82 85 92
Email: info.fr@altium.com

Altium™

www.tasking.com