

RVCT

From [The RealView Compilation Tools Web Page at ARM.com](#):

The ARM® RealView® compilation tools have been researched and developed for over 16 years to provide optimum support for the ARM architecture. The compilation tools deliver the full set of software components required to build C and C++ applications targeting the 32-bit ARM and 16-bit Thumb® and Thumb-2 instruction sets.

The compilation tools in RealView Development Suite are developed directly with new ARM architectures enabling specific optimizations for each of these ARM architectures and delivering significant improvements in code speed and size.

GNU Interoperability Create highly optimized applications for Embedded Linux and Symbian OS using the compilation tools in RealView Development Suite. RealView Development Suite provides unprecedented ease-of-use for customers seeking interoperability between ARM and GNU toolchains that are compatible with the Application Binary Interface (ABI) for the ARM architecture, allowing flexible deployment of open-source and commercially supported tools throughout software development teams.

The compilation tools in RealView Development Suite include:

Optimizing ISO C Compiler Optimizing ISO C++ Compiler Linker Assembler Image Conversion Tool ARM Object File Librarian/Archiver C Libraries RogueWave C++ Standard Template Libraries Optimizing ISO C/C++ Compiler and Assembler

Main Features:

Full ISO C and C++ Support Implementation of ARM C/C++ ABI for the ARM Architecture Enables mixing of object code compiled with other ABI compliant compilers, e.g. GNU Industry-leading code size optimization Industry-leading code performance optimization Compilation for 32-bit ARM and 16-bit Thumb and Thumb-2 instruction sets Selectable debug and optimization levels Processor-specific optimizations for all ARM architectures Object files conform to industry standard ELF and DWARF Powerful macro assembler for ARM, Thumb and Thumb-2 instructions Linker

ELF and DWARF industry standards supported for object file and debug table formats Seamless interworking of ARM Thumb and Thumb-2 object code Scatter-loading feature supports placement of code and data within sophisticated target memory maps Automatic removal of unreferenced code areas Image Conversion Tool

Converts from ELF images into other downloadable and ROMable formats Binary, Motorola 32-bit S-record, Intel Hex-32 and Byte Oriented Hex formats are supported Displays information about the input file, e.g. disassembly output or symbol listings ARM Object File Librarian/Archiver

Enables sets of ELF object files to be collected together and maintained in libraries Supports merging of libraries C and RogueWaveC++ Libraries

The full ISO standard C libraries consist of: Functions defined by the ISO C library standard Target-dependent functions used to implement the C library functions in the semihosted execution environment Helper functions used by the C and C++ compilers Target-dependent C library functions can be re-implemented for any execution environment (semihosting) The floating-point library uses the ARM floating-point environment, which is an implementation of the IEEE 754 standard for binary floating-point arithmetic The Rogue Wave C++ Standard Template Library contains: The functions defined by the ISO C++ library standard The Rogue Wave Standard C++ Library version 2.02.03 Helper functions for the C++ compiler Semihosting Support Semihosting is the mechanism by which ARM targets communicate

RVCT

input/output requests from application code to a host computer running a debugger. This mechanism could be used, for example, to allow functions in the C library, such as `printf()` and `scanf()`, to use the screen and keyboard of the host rather than the target system's resources.

Internal links

- [GCC](#)
- [GCCE](#)
- [Microsoft C++ compiler](#)
- [Nokia x86 compiler](#)