

A Debugger on Every Desk

A DEBUGGER FOR EVERY DESKTOP

Rather than purchase a few high-end debuggers that developers have to share, a better solution is fewer expensive debuggers along with several streamlined debuggers that can analyze the majority of issues.

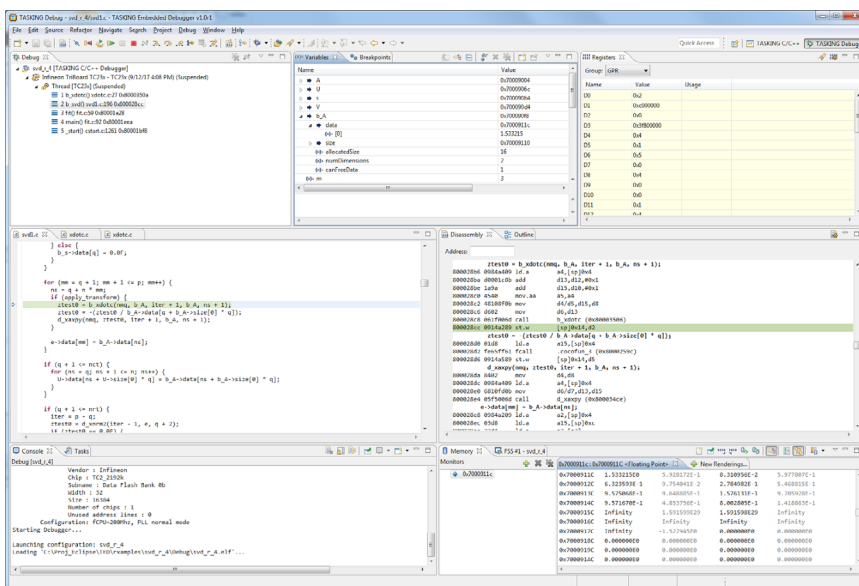
High-end debuggers are expensive primarily because of the expensive hardware probes with deep trace memories and ability to run complex timing analysis. However, most of the debugging time spent by developers is not with complex issues, but rather simple code verification that does not require these expensive features.

The TASKING Embedded Debugger is just that solution. It provides robust debugging along with inexpensive hardware probes and simulators to allow developers to quickly analyze and debug code, keeping their “coding momentum.” Common debugging functions such as program download, execution control, breakpoints, C/C++ variable access, and multi-core support, are all available.

Interrupting the developers edit-compile-debug cycle to wait for a debugger license is not only inefficient, but developers usually develop a “momentum” when coding...a string of contiguous thought about how to solve the problem. If this is interrupted, it can take time to regain that train of thought. Enough TASKING Embedded Debuggers can be deployed to keep coding efficient and maintain coding momentum.

DEBUG FUNCTIONS AND FEATURES

The debugger is an ECLIPSE based stand-alone tool, or can be integrated as a plugin into existing Eclipse Mars GUIs. It uses the on-chip debug features of the physical target, or connects to an instruction-set simulator.



The GUI is familiar Eclipse-based and simple to use. This view shows stack of active thread, C and assembly code, value of variables and registers, and the memory formatted as floating point values.

Product Benefits

- Economical - Allows many to be purchased for the cost of one high-end debug solution
- Fast, efficient debugging - lets developers verify code functionality and get back to coding quickly
- Comprehensive support for Infineon Tricore devices running highly optimized binaries created by TASKING toolsets v4.2r2 and higher
- Performs all common debugging functions, for all cores on Tricore AURIX devices
- Using low cost probes and instruction-set simulators
- Most cost-effective debug solution ideally suited for large development teams

Debug Functions and Features

- ECLIPSE based
- OCDS/MCDS hardware debug
- Ample board support
- Fast download to RAM and FLASH
- C/C++ and Assembly support
- Debugging of highly optimized code
- C/C++ expression evaluation
- Symbolic SFR access
- Multi-core start/stop synchronization
- On-chip breakpoints and watch-points
- Unlimited number of breakpoints in RAM
- RTOS aware debugging
- Basic trace support
- File system virtualization
- Script based testing

Board support is provided for evaluation boards from Infineon and 3rd-parties. Programs can be downloaded into RAM and FLASH memory and debugged at C/C++ and/or Assembly level. All variable types specific to the high-level language can be displayed, modified and used in expressions, and also peripheral register settings can be accessed symbolically. Elaborate run-control features are available including interrupt-aware stepping and multi-core start/stop synchronization. A limited number of hardware breakpoints and watchpoints as well as an unlimited number of software breakpoints are available to (conditionally) stop program execution. RTOS aware debugging is supported via the OSEK Run Time Interface (ORTI), and basic trace support is available on emulation devices and production devices equipped with miniMCDS. Remote debugging is supported for boards connected via DAS. Hence the developer can easily connect the debugger to a physical target device via a network connection. This way, the device can be time-shared among multiple developers. Debug and test flows can be automated using the available script language, and the virtual file system simulation capability offers a means to easily move data from the board to the host system for further analysis.

SUMMARY

The TASKING Standalone Debugger is a cost-efficient, yet complete, solution for code verification in large development teams. For the price of a single high-end debugger, with functionality that is only required for a small percentage of the time, several streamlined debuggers can be purchased, enabling developers to identify logic problems and correct coding errors early on.

Probes and Targets

- Supported TriCore Devices:
 - TC11xx Family (TC1130, TC1164, TC1166, TC1167, TC1197)
 - TC173x Family (TC1736)
 - TC176x Family (TC1762, TC1766, TC1767)
 - TC178x Family (TC1782, TC1784)
 - TC179x Family (TC1791, TC1792, TC1793, TC1796, TC1797, TC1798)
 - AURIX TC2xx Family (TC21X, TC22X, TC23X, TC26X, TC27X, TC29X)
 - AURIX TC3xx Family
- Supported Debug Probes
 - Infineon Device Access Server (DAS) v6.0 and DAP miniWiggler
- Supported Single-core Instruction Set Simulators
 - TriCore, GTM/MCS, PCP, HSM and XC800 (all included in package)