

# Leverage Freertos for MSP432 Mcus



The use of an ARM Cortex-M4 core in the new MSP432 microcontroller platform has brought the benefits of the huge, diverse and non-proprietary ARM ecosystem to MSP430 MCU users. One of the most popular components in that ecosystem is [FreeRTOS](#) – probably the most widely used and best-supported real time kernel [in its class] on the planet. Real Time Engineers Ltd. released [a maintained FreeRTOS demo](#) for MSP432 MCUs on the day TI’s new microcontroller platform was announced.

The FreeRTOS demo targets the MSP-EXP432P401R LaunchPad Development Kit, and pre-configured build projects are provided for both the IAR and Keil tools – two of the most popular mid-priced development environments for ARM microcontrollers. Also, and for the first time, there is now a FreeRTOS Cortex-M port for TI’s own ARM compiler, along with a maintained demo for Code Composer Studio (CCS) IDE that uses the new port. FreeRTOS’s GCC and Tasking compiler ports for the Cortex-M4F will also work with MSP432 MCUs, although no pre-configured MSP432 MCU examples are provided for those compilers yet.

Updated FreeRTOS demos have also been created for, and will be maintained for, MSP430FR5969 MCUs. At the time of writing, the MSP430FR5969 MCU demo is only available from FreeRTOS’s public SVN repository, but it will be included in the next official FreeRTOS release.

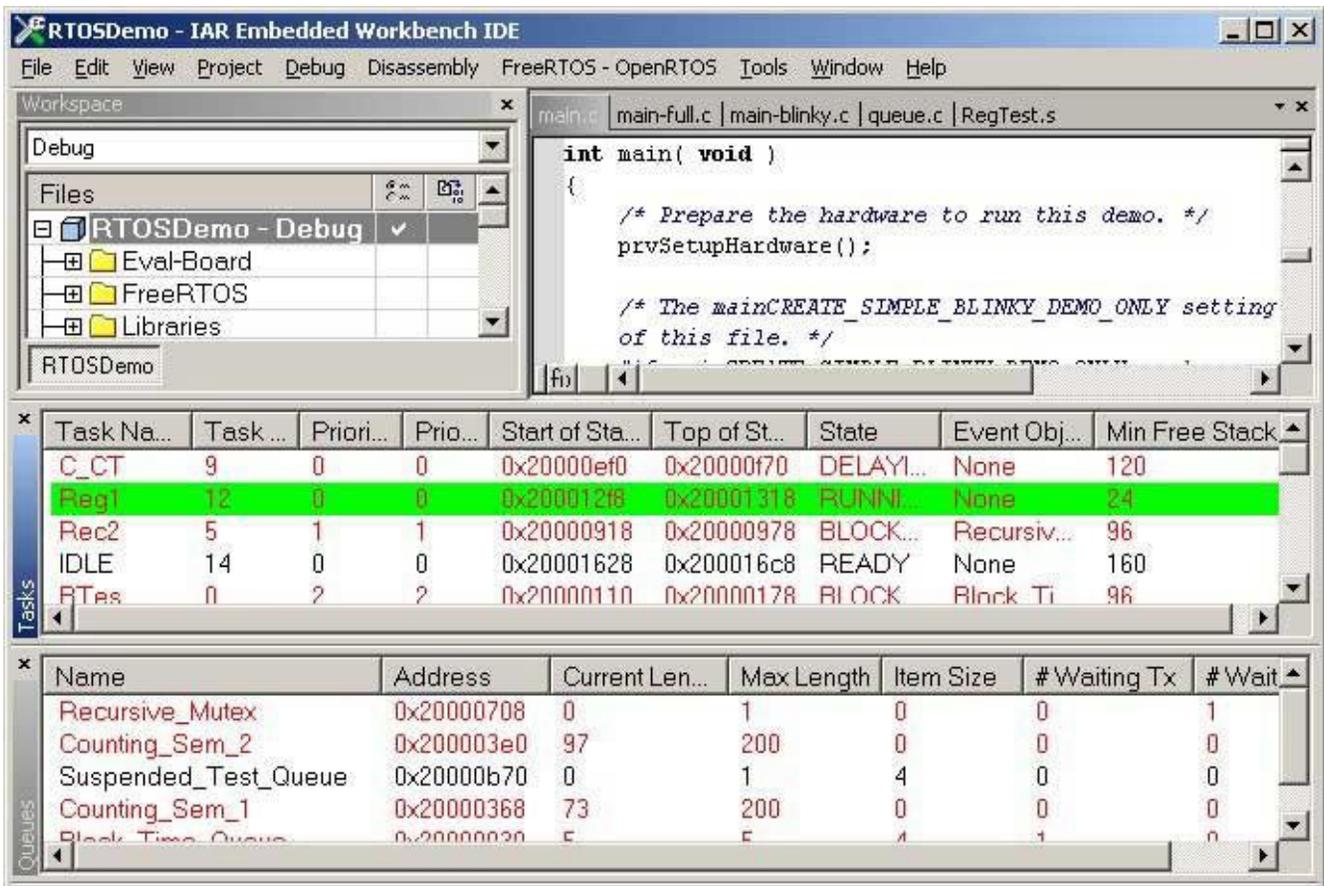


Figure 1. The FreeRTOS StateViewer plug-in shipped with the IAR IDE

For the power-consumption-conscious, FreeRTOS has long included an option for tick suppression through its tick-less idle mode. Tick-less idling allows the tick interrupt to be turned off only when it is pre-determined that

---

no tasks are scheduled to run for a fixed and user-configurable time into the future – typically three to ten ticks into the future depending on the time and energy required to enter then exit a deep sleep mode. Turning the tick interrupt off only during idle periods removes the need for the next task execution time to be determined, and then the clock re-programmed appropriately, on each context switch – reducing the processing overhead and the length of each task switch. Pre and post sleep hook functions enable calls into application specific sleep management libraries in order to determine the sleep mode that should be entered, or even to abort entry into a sleep mode at altogether (note the initial FreeRTOS demo uses the generic Cortex-M tick-less mode, rather than an implementation tailored specifically to use the low power features of MSP432 MCUs).

FreeRTOS has become the globally successful de facto standard for microcontrollers by offering a truly compelling value proposition, with complete peace of mind; it is professionally developed, strictly quality controlled, robust, supported, independent, cross platform, and free to embed in commercial products without any requirement to expose your proprietary source code. It has also removed all the common objections to using free software in commercial applications, including the risk of accidental IP infringement. Even so, some companies still insist that somebody is accountable for all the software that is incorporated into their products, and for those customers a very low-cost commercial license for FreeRTOS may be purchased from our partner company. Commercially licensed versions of FreeRTOS are fully indemnified, and sold under the OpenRTOS brand.

## IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to [TI's Terms of Sale](#) or other applicable terms available either on [ti.com](https://www.ti.com) or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265  
Copyright © 2023, Texas Instruments Incorporated