Technical Article
Enabling Wi-Fi® and Bluetooth® Connectivity on RTOS

TEXAS INSTRUMENTS

Prajakta Desai

Trish Messiter

Have You Ever Thought That It Would Be Nice to Maintain Just One Wireless Application Even Though You Have Multiple Target Hardware/software (MCU/MPU; OS/RTOS)?

In the busyness that characterizes today's product development and maintenance cycles, is there really time to deal with the lower level details at the embedded application layer? The time to develop is lessening whilst simultaneously the feature set and complexity of requirements expands. This means that more must be done in less time, so busy engineers need ways to perform their tasks with increasing efficiency.

One part of the increasing complexity of designs is the addition of short range wireless connectivity to many product categories that, until recently, was thought to be unnecessary. Engineers now often have to deal not only new designs but with updating existing product families to add wireless capabilities. Often the different members of the product families have differences in hardware (MCU/MPU, wireless chipset) and/or in software (OS/RTOS, TCP/IP stack).

This mixture of issues prompted Clarinox to develop middleware to abstract wireless protocol stack software so that the stacks can remain consistent across multiple hardware and software. The wireless application written for these stacks can remain the same across multiple hardware and software. This reduces the development and maintenance task down to a single application, as opposed to the difficulties in debugging different wireless performance on each different target.
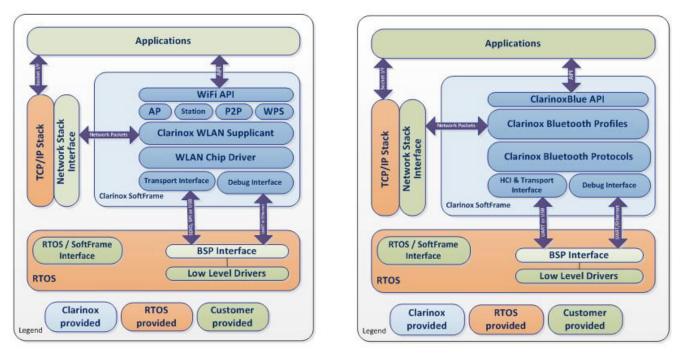
Short range wireless is fundamental to the Internet of Things (IoT), which despite the nay-sayers is unlikely to disappear any time soon. In fact, by most predictions, it is likely to be the most significant area of market growth of this decade.

Along with the need for wireless, it makes sense to use RTOS for the efficiencies gained in the management of intertask communication and resource sharing; such memory allocation, interrupt handlers, semaphores and the like. But which RTOS? There are many and each has strengths and weaknesses. An RTOS may be chosen for legacy reasons or may be selected for its applicability to a specific set of requirements. Whichever is the case, the Clarinox middleware enables this level of choice. Application code developed for one target can be reused on another. Saving valuable time in development and simplifying maintenance.

Below picture shows how this is achieved for *Bluetooth*® and Wi-Fi®.

The ClarinoxBlue Bluetooth Classic and Bluetooth Low Energy protocol stack along with the ClarinoxWiFi WLAN protocol stack are encapsulated within the ClarinoxSoftFrame middleware. This middleware enables the SAME stacks to run across eCos, embOS, FreeRTOS, Green Hills INTEGRITY, Linux®/ Embedded Linux, Microsoft (Windows, Windows CE, & Windows Phone), MQX, Nucleus by Mentor Graphics, QNX, RTX, TI-RTOS Kernel, formally known as SYS/BIOS platform, Express Logic ThreadX, uC/OS-II and uC/OS-III platforms by Micrium, VxWorks by Wind River® Systems.

For most of these RTOS the hardware supported via the RTOS BSP quickly enables support by Clarinox. Clarinox MCU/MPU experience includes TI's DM6435 and DM3730 processors, Sitara™ AM335x processors, OMAP-L138, TMS320C6000 (digital signal processor) DSP + ARM® processors and TMS320C6748 DSPs, TM4C129x as well as a wide variety of MCUs and processors outside of TI's portfolio.

Clarinox has supported multiple Bluetooth chipsets and for Bluetooth and Wi-Fi combinations including TI's WiLink™ 8 modules. Clarinox customers using a WiLink 8-based connectivity solution include SMART Technologies, Navico and many more!

So next time you think it would be nice to maintain just one wireless application across multiple target hardware/ software (MCU/MPU; OS/RTOS), contact Clarinox to show you how.

IMPORTANT NOTICE AND DISCLAIMER