# PREFACE

This book is intended to provide the practicing engineer with the necessary background to apply real-time theory to the design of embedded components and systems in order to successfully field a real-time embedded system. The book also is intended to provide a senior-year undergraduate or first-year graduate student in electrical engineering, computer science, or related fields of study with a balance of fundamental theory, review of industry practice, and hands-on experience to prepare for a career in the real-time embedded system industries. Typical industries include aerospace, medical diagnostic and therapeutic systems, telecommunications, automotive, robotics, industrial process control, media systems, computer gaming, and electronic entertainment, as well as multimedia applications for general-purpose computing. Real-time systems have traditionally been fielded as hard real-time applications, such as digital flight control systems, antilock braking systems, and missile guidance. More recently, however, intense interest in soft real-time systems has arisen due to the quickly growing market for real-time digital media services and systems.

This updated edition adds three new chapters focused on key technology advancements in embedded systems and with wider coverage of real-time architectures. The overall focus remains the RTOS (Real-Time Operating System), but use of Linux for soft real-time, hybrid FPGA (Field Programmable Gate Array) architectures and advancements in multi-core system-on-chip (SoC), as well as software strategies for asymmetric and symmetric multiprocessing (AMP and SMP) relevant to real-time embedded systems, has been added. Specifically, a new Chapter 9 provides an overview of RTOS advancements, including AMP and SMP configurations, with a discussion of future directions for RTOS use in multi-core architectures, such as SoC. A new Chapter 10 is devoted to open source RTOS, with emphasis on FreeRTOS. A new Chapter 11 is focused on methods to integrate embedded Linux into real-time embedded systems, with emphasis on soft real-time requirements, methods to patch and improve the Linux kernel for predictable response, and finally best practices for implementation of real-time services and applications that make use of POSIX real-time extensions in the 1003.1 2013 standard. The original Chapters 9, 10, and 11 have been preserved and are now Chapters 12 to 14, and Part III remains unchanged other than chapter renumbering to accommodate the insertion of the new chapters.

John Pratt, a new co-author, has contributed extensively to this edition, with specific focus on FreeRTOS, and brings a unique perspective to this updated version with his commercial mobile embedded systems expertise.

The new Linux examples and extended coverage of Linux in this edition are based upon a summer version of the course *Real-Time Embedded Systems* taught at the University of Colorado, Boulder, to offer an alternative to the traditional fall course that has used the Wind River VxWorks RTOS. The summer course has emphasized the same hard and soft real-time theory, but practice has focused on using Linux to achieve predictable response for systems that require real-time, but where occasional missed deadlines are not catastrophic. For example, mobile digital media, augmented reality applications, computer vision and digital entertainment and interactive systems. While hard real-time mission critical systems are still a major concern, many emergent applications require predictable response and simply need to provide high-quality of service. The use of buffering and time stamps to work ahead and provide high-quality presentation of results is, for example, a method used in digital video encode, transport, and decode, where the systems software is not required to provide deterministic proven hard real-time processing, as has been the goal for the RTOS. Likewise, many systems today use best-effort or soft real-time embedded Linux configurations with coprocessors, either FPGA or ASIC (Application Specific Integrated Circuits), that provide guaranteed hard real-time processing with hardware state machines.

Numerous improvements and corrections have been made to the original edition text to improve readability and clarity based on excellent feedback by many undergraduate and graduate students at the University of Colorado who have used the original edition text since August 2006.

While it's impossible to keep up with all the advancements related to real-time embedded systems, we hope the reader will find the new chapters and expanded example material included on the DVD a useful extension to traditional cyclic executive and RTOS real-time system components and systems architecture. The expanded guidelines and strategies are intended to help the practicing engineer and to introduce advanced undergraduate and graduate students in computer and software engineering disciplines to design strategies and methods to tackle many of the challenges found in real-time embedded systems. This challenging engineering area continues to evolve and relies upon the careful validation and verification efforts of practicing engineers to ensure and balance safety, cost, and capabilities of these complex and critical applications on which we all depend.

## Companion Files

Companion files (figures and code listings) for this title are also available by contacting info@merclearning.com.

Sam Siewert
December 2015