lead to disaster. Thus, this third edition includes a new chapter on *cloud and fog computing* that argues for the *fog architectural style* to decouple the *world of cloud* from the *world of real-time embedded systems*. The second major change is the rewritten chapter on *real-time networks* that now also covers IEEE 802.1 Time-Sensitive Networking (TSN), the incorporation of the time-triggered paradigm in the mainstream IT networking set of standards. All other chapters have been revised and updated. This third edition includes approximately 80 new references in total.

Since the publication of the first edition, a visible paradigm shift from the event-triggered to the time-triggered design methodology for dependable distributed real-time systems has taken place in a number of applications.

It is assumed that the reader of this book has a background in basic computer science or computer engineering or has some practical experience in the design or implementation of embedded systems.

The glossary, provided at the end, is an integral part of this book, providing definitions for many of the technical terms used throughout. If the reader is not sure about the meaning of a term, they are advised to refer to the glossary.

Acknowledgments

It is impossible to name all students, colleagues from industry, and fellow scientists who have contributed to this third edition of the book by asking intelligent questions or making constructive comments over the last decade—thanks to all of you.

Vienna, Austria — Hermann Kopetz, Wilfried Steiner

May 2022

Contents

Chapter 1
The Real-Time Environment

Overview

The purpose of this introductory chapter is to describe the environment of real-time computer systems from a number of different perspectives. A solid understanding of the technical and economic factors that characterize a real-time application helps to interpret the demands that the system designer must cope with. The chapter starts with the definition of a real-time system and with a discussion of its functional and nonfunctional requirements. Particular emphasis is placed on the temporal requirements that are derived from the well-understood properties of control applications. The objective of a control algorithm is to drive a process such that a performance criterion is satisfied. Random disturbances occurring in the environment degrade system performance and must be taken into account by the control algorithm. Any additional uncertainty that is introduced into the control loop by the control system itself, e.g., a non-predictable jitter of the control loop, results in a degradation of the quality of control.

In Sects. 1.2, 1.3, 1.4 and 1.5, real-time applications are classified from a number of viewpoints. Special emphasis is placed on the fundamental differences between *hard* and *soft* real-time systems. Because soft real-time systems do not have severe failure modes, a less rigorous approach to their design is often followed. Sometimes resource-inadequate solutions that will not handle the rarely occurring peak-load scenarios are accepted on economic arguments. In a hard real-time application, such an approach is unacceptable because the safety of a design in all specified situations, even if they occur only very rarely, must be demonstrated vis-a-vis a certification agency. In Sect. 1.6, a brief analysis of the real-time system market is carried out with emphasis on the field of embedded real-time systems. An embedded real-time system is a part of a self-contained product, e.g., a television set or an automobile. Embedded real-time systems, also called *cyber-physical systems* (CPS), form the most important market segment for real-time technology and the computer industry in general.

H. Kopetz, W. Steiner, *Real-Time Systems*,
https://doi.org/10.1007/978-3-031-11992-7_1