TUHH
Hamburg University of Technology

Institute of Embedded Systems

Compilers for Embedded Systems

Summer Term 2022

Heiko Falk

Institute of Embedded Systems
Electrical Engineering, Computer Science and Mathematics
Hamburg University of Technology

TUHH
Hamburg University of Technology

Institute of Embedded Systems

Chapter 1

Introduction & Motivation

Embedded Systems

Definition: Embedded Systems (ES) are
- information-processing systems
- that are embedded into a larger, surrounding product.

- The fact that embedded systems process information is not decisive for a purchase
- Instead: Benefit of surrounding product influences purchase decision

*[**P. Marwedel.** Embedded System Design. Springer, 2011]*

Application Areas of Embedded Systems

Consumer products

Multimedia

Transportation

Telecommunication

Building management, robotics, …

Relevance of Embedded Systems

- **Smartphones** 113 m. devices 2007 ⇝ *25.6% increase p.a.*
  365 m. devices 2012
- **UMTS** 402 m. customers worldwide 2008
  30 m. new customers per quarter
- **Energy consumption of mobile broadband infrastructure**
  42.8 m. kWh 2005 ⇝ 124.4 m. kWh 2011
- **Broadband internet**
  576 m. customers 2011 ⇝ *100% increase compared to 2007*
- **US consumer electronics**
  ∅ household: 25 devices, ∅ *adult: 1,200$ p.a.*

[www.itfacts.biz]

Demands on Embedded Systems (1)

Efficiency

– Runtime efficiency

– Energy consumption

– Code size

– Physical size / weight

– Cost

Demands on Embedded Systems (2)

Real-Time Capabilities

- **Definition:** For some input *x*, a computer system computes some *f(x)*.
  - A ***Non-Real-Time System*** is correct if it computes *f(x)* correctly.
  - A ***Real-Time System*** is correct if the computation of *f(x)* completes *within given time bounds*.

☞ **A too late computation of *f(x)* by a Real-Time System is as severe as a functionally incorrect computation.**

Demands on Embedded Systems (3)

Real-Time Capabilities

– **“Hard” Real-Time System:**
Late computation of *f(x)* ↝ Catastrophic consequences
(loss of human life, environmental damages, …)

Example airbag control:
Command to ignite the airbags: 15ms
Deadline miss: Danger of injuries for passengers or saviors. Thus: Do not fire airbags

– **“Soft” Real-Time System:** No catastrophic consequences
Example DVD player:
Deadline miss during frame decoding: *Frame-Drop*
Not nice but (usually) not disastrous

Demands on Embedded Systems (4)

Dependability / Safeness

– Life span of embedded systems: Typically several years
– During the entire lifetime: No outage

– *Example throttle valve control:*

Production volume:	2 m. units per year
Allowed error rate:	1 unit per year

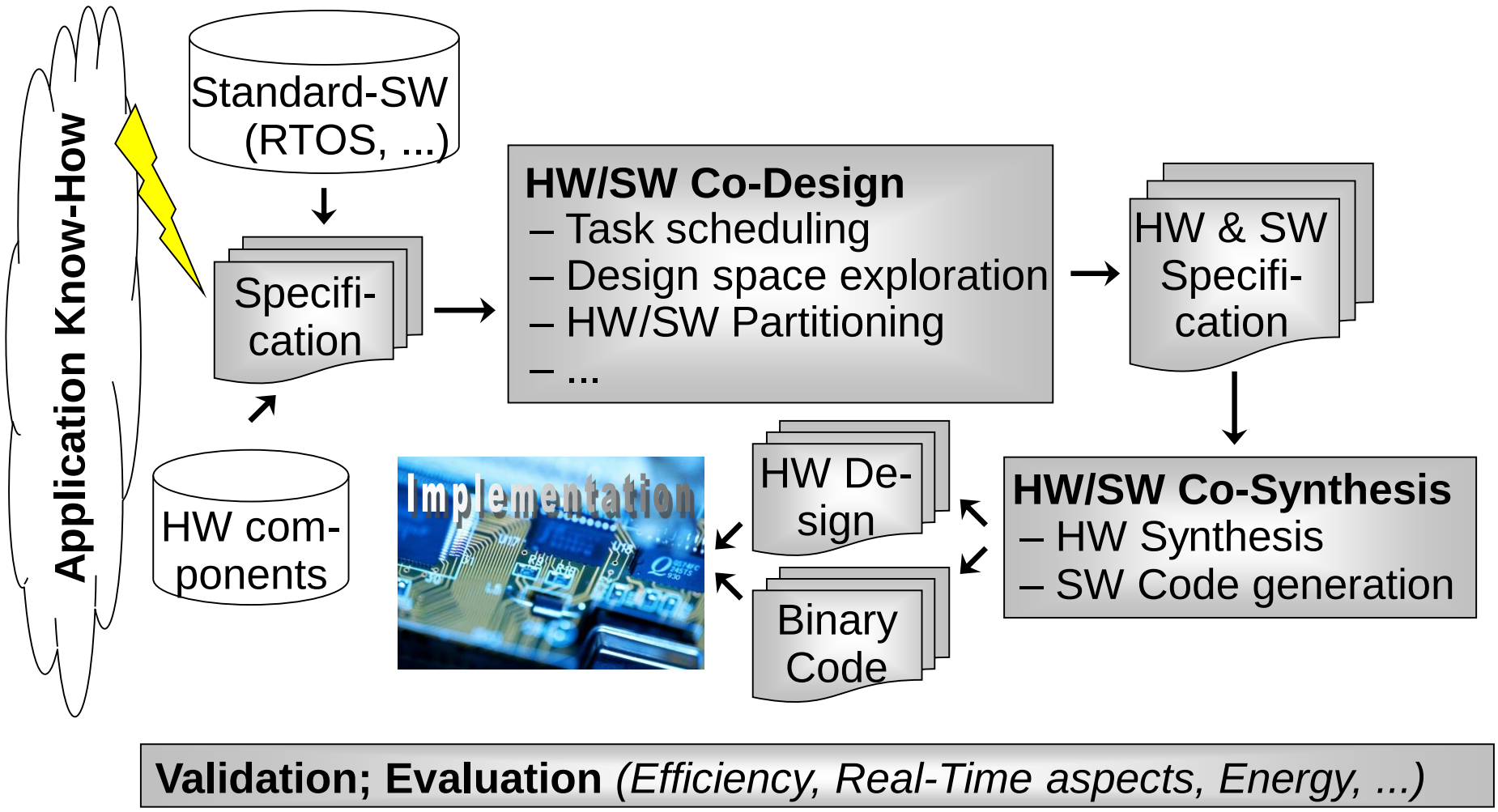
Maintainability, (Limited Extensibility)

– Error search and diagnosis, reconfiguration during runtime

Design Automation *(↝ Time to Market)*

– Specification, synthesis, code generation

Design Process of Embedded Systems



Topics of the Course

Overview

1. Introduction & Motivation
2. Compilers for Embedded Systems – Requirements & Dependencies
3. Internal Structure of Compilers
4. Pre-Pass Optimizations
5. HIR Optimizations and Transformations
6. Code Generation
7. LIR Optimizations and Transformations
8. Register Allocation
9. WCET-Aware Compilation
10. Outlook

General References

Embedded Systems

– Peter Marwedel. *Embedded System Design*. Springer, 2011.
ISBN 978-94-007-0256-1
Full-text available as e-book via TUHH library:
https://katalog.tub.tuhh.de/Record/1650607989

Compiler Construction

– Steven S. Muchnick. *Advanced Compiler Design & Implementation*. Morgan Kaufmann, 1997.
ISBN 978-1-55860-320-2
– Andrew W. Appel. *Modern compiler implementation in C*. Cambridge University Press, 2004.
ISBN 0-521-60765-5