



Universiteit  
Leiden  
The Netherlands

## Latency, energy, and schedulability of real-time embedded systems

Liu, D.; Liu D.

### Citation

Liu, D. (2017, September 6). *Latency, energy, and schedulability of real-time embedded systems*. Retrieved from <https://hdl.handle.net/1887/54951>

Version: Not Applicable (or Unknown)

License: [Licence agreement concerning inclusion of doctoral thesis in the Institutional Repository of the University of Leiden](#)

Downloaded from: <https://hdl.handle.net/1887/54951>

**Note:** To cite this publication please use the final published version (if applicable).

Cover Page



Universiteit Leiden



The handle <http://hdl.handle.net/1887/54951> holds various files of this Leiden University dissertation

**Author:** Liu, D.

**Title:** Latency, energy, and schedulability of real-time embedded systems

**Issue Date:** 2017-09-06

# **Latency, Energy, and Schedulability of Real-Time Embedded Systems**

Di Liu



# **Latency, Energy, and Schedulability of Real-Time Embedded Systems**

## **PROEFSCHRIFT**

ter verkrijging van  
de graad van Doctor aan de Universiteit Leiden,  
op gezag van Rector Magnificus Prof.mr. C.J.J.M. Stolker,  
volgens besluit van het College voor Promoties  
te verdedigen op woensdag 6 september 2017 klokke 13:45 uur

door

Di Liu  
geboren te Lijiang, China  
in 1984

<b>Promotor:</b>	Prof. Dr. Joost N. Kok	Universiteit Leiden
<b>Co-Promotor:</b>	Dr. Todor P. Stefanov	Universiteit Leiden
<b>Promotion Committee:</b>	Prof. Dr. Koen Langendoen Prof. Dr. Kai Huang Dr. Sebastian Altmeier Prof. Dr. Aske Plaat Prof. Dr. Thomas Bäck Prof. Dr. Harry A.G. Wijshoff	Technische Universiteit Delft Sun Yat-Sen University, China Universiteit van Amsterdam Universiteit Leiden Universiteit Leiden Universiteit Leiden

Latency, Energy, and Schedulability of Real-Time Embedded Systems

Di Liu. -

Dissertation Universiteit Leiden. - With ref. - With summary in Dutch.

Copyright © 2017 by Di Liu. All rights reserved.

This dissertation was typeset using L<sup>A</sup>T<sub>E</sub>X in Linux and version controlled using Git.

ISBN: 978-94-6299-658-8

Printed by: Ridderprint, the Netherlands.

# Contents

<b>Contents</b>	<b>v</b>
<b>List of Tables</b>	<b>ix</b>
<b>List of Figures</b>	<b>xi</b>
<b>List of Abbreviations</b>	<b>xiii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Development Trends in Real-Time Embedded Systems . . . . .	2
1.1.1 The Era of Multicore/Multiprocessor Systems . . . . .	3
1.1.2 The Shift to Heterogeneous Multicore Systems . . . . .	4
1.1.3 The Emergence of Mixed-Criticality Systems . . . . .	5
1.2 Problem Statement . . . . .	6
1.3 Contributions of This Dissertation . . . . .	9
1.4 Dissertation Outline . . . . .	11
<b>2 Background</b>	<b>13</b>
2.1 Cyclo-Static Dataflow (CSDF) Model . . . . .	13
2.2 Real-Time Theories . . . . .	15
2.2.1 Real-Time Task Models . . . . .	16
2.2.2 Real-Time Scheduling . . . . .	17
2.2.3 Multiprocessor Real-Time Scheduling . . . . .	20
2.3 Hard-Real-Time (HRT) Scheduling of CSDF graphs . . . . .	23
<b>3 Resource Optimization for Real-Time Streaming Application</b>	<b>27</b>
3.1 Background . . . . .	28
3.2 Related Work . . . . .	29
3.3 Motivational Example . . . . .	30
3.4 Proposed Approach . . . . .	30

3.5	Evaluation . . . . .	34
3.5.1	The effectiveness of our DM approach . . . . .	35
3.5.2	The time complexity of solving our DM problem . . . . .	36
3.6	Discussion . . . . .	38
<b>4</b>	<b>Energy Optimization for Real-Time Streaming Applications</b>	<b>41</b>
4.1	Related Work . . . . .	42
4.2	Background . . . . .	43
4.2.1	System Model . . . . .	43
4.2.2	Energy Model . . . . .	45
4.3	Proposed Mapping Algorithm . . . . .	48
4.3.1	Processor Type Assignment . . . . .	49
4.3.2	Task mapping . . . . .	52
4.3.3	Remapping . . . . .	53
4.3.4	The FDM Algorithm . . . . .	58
4.4	Evaluation . . . . .	59
<b>5</b>	<b>Energy Optimization for Real-Time Tasks</b>	<b>67</b>
5.1	Related Work . . . . .	68
5.2	Background . . . . .	70
5.2.1	System Model . . . . .	70
5.2.2	Task Model . . . . .	70
5.2.3	Energy Model . . . . .	71
5.2.4	C=D Task-Splitting . . . . .	71
5.3	Motivational Example . . . . .	72
5.4	C=D Task-Splitting on Heterogeneous Multiprocessor Systems . . . . .	74
5.4.1	Task Splitting . . . . .	74
5.4.2	Subtask Allocation . . . . .	75
5.5	Allocation and Split on Heterogeneous Multicore Systems (ASHM) . . . . .	76
5.5.1	Allocation and splitting of E-tasks . . . . .	77
5.5.2	Allocation and Splitting of NE-tasks . . . . .	78
5.5.3	The SPLIT function . . . . .	80
5.5.4	Computing the minimum frequency . . . . .	83
5.5.5	The ASHM Algorithm . . . . .	84
5.6	Evaluation . . . . .	85
5.6.1	Experimental Setup . . . . .	86
5.6.2	Experimental Results . . . . .	87
5.7	Discussion . . . . .	91

<b>6 Schedulability Analysis of Imprecise Mixed-Criticality Systems</b>	<b>93</b>
6.1 Related Work . . . . .	94
6.2 Preliminaries . . . . .	95
6.2.1 Imprecise Mixed-Criticality Task Model . . . . .	95
6.2.2 Execution Semantics of the IMC Model . . . . .	97
6.2.3 EDF-VD Scheduling . . . . .	97
6.2.4 An Illustrative Example . . . . .	98
6.3 Schedulability Analysis . . . . .	98
6.3.1 Low Criticality Mode . . . . .	98
6.3.2 High Criticality Mode . . . . .	99
6.4 Speedup Factor . . . . .	105
6.5 Experimental Evaluation . . . . .	112
6.5.1 Comparison with AMC [BB13] . . . . .	113
6.5.2 Impact of $\alpha$ and $\lambda$ . . . . .	115
<b>7 Summary and Future Work</b>	<b>117</b>
7.1 Summary and Conclusions . . . . .	117
7.2 Future work . . . . .	119
7.2.1 The real convergence of data-flow models and real-time theories	119
7.2.2 The multi-objective mapping of heterogeneous multicore systems . . . . .	120
7.2.3 Practical and flexible MC model . . . . .	120
<b>Bibliography</b>	<b>121</b>
<b>Appendix</b>	<b>129</b>
<b>List of Publications</b>	<b>135</b>
<b>Index</b>	<b>137</b>
<b>Samenvatting</b>	<b>138</b>
<b>Acknowledgements</b>	<b>141</b>
<b>Curriculum Vita</b>	<b>143</b>



# List of Tables

1.1	Criticality levels in the DO-178B/C standard [Nor] . . . . .	3
1.2	Comparison between ARM Cortex A57 and ARM Cortex A53 [ARM16]	5
2.1	Task models considered in the dissertation' chapters . . . . .	17
2.2	Scheduling algorithms considered in each chapter . . . . .	22
3.1	Tasks Parameters 1 . . . . .	31
3.2	Tasks Parameters 2 . . . . .	31
3.3	Characteristics of application benchmarks . . . . .	35
3.4	Latency Constraints . . . . .	35
3.5	The execution time of our DM approach (in second) . . . . .	38
4.1	The difference from [KYD11] and [CKR14] . . . . .	43
4.2	The 'uncore' power consumption . . . . .	46
4.3	The estimated parameters . . . . .	46
4.4	The parameters of H.263 . . . . .	49
4.5	Different processor type assignments for the H.263 decoder . . . . .	50
4.6	Different mappings for H.263 decoder . . . . .	52
4.7	The Streaming Applications . . . . .	61
4.8	Cluster Heterogeneous MPSoC configurations . . . . .	61
4.9	Summary of Figure 4.5 . . . . .	62
4.10	Summary of Figure 4.6 . . . . .	63
4.11	Summary of Figure 4.7 . . . . .	64
5.1	Power parameters for different core types . . . . .	72
5.2	The original task set . . . . .	72
5.3	Split subtasks . . . . .	73
5.4	Energy consumption . . . . .	73
5.5	Split Example . . . . .	74

6.1 Illustrative example . . . . .	98
6.2 The speedup factor w.r.t $\alpha$ and $\lambda$ . . . . .	111

# List of Figures

2.1	CSDF graph $G$	15
3.1	Production and consumption curves on edge $e_u = (\tau_i, \tau_j)$	33
3.2	Global scheduling with $L_0$ constraint	36
3.3	Global scheduling with $L_1$ constraint	37
3.4	Global scheduling with $L_2$ constraint	37
4.1	An example of a cluster heterogeneous MPSoC	44
4.2	Power model validation of PE (big) cluster	47
4.3	Power model validation of EE (LITTLE) cluster	48
4.4	H.263 Decoder	49
4.5	Comparison between FDM and CKR	62
4.6	FDM vs. Algorithm 2+KYD	63
4.7	FDM vs. KYD on homogeneous MPSoCs	64
5.1	Varying $U$ on platform with 2 PE cores and 2 EE cores	89
5.2	Varying $U$ on platform with 2 PE cores and 3 EE cores	89
5.3	Varying $U$ on platform with 3 PE cores and 2 EE cores	89
5.4	Varying the number of tasks on platform with 2 PE cores and 2 EE cores	90
5.5	Varying the number of tasks on platform with 2 PE cores and 3 EE cores	90
5.6	Varying the number of tasks on platform with 3 PE cores and 2 EE cores	90
6.1	Scheduling of Example 6.1	98
6.2	plane 1	107
6.3	plane 2	108
6.4	vertical surface	108
6.5	3D space of optimization problem (6.17)	109

6.6	3D image of the speedup factor w.r.t $\alpha$ and $\lambda$ . . . . .	111
6.7	$\lambda = 0.3$ . . . . .	114
6.8	$\lambda = 0.5$ . . . . .	114
6.9	$\lambda = 0.7$ . . . . .	114
6.10	Impact of $\lambda$ . . . . .	115
6.11	Impact of $\alpha$ . . . . .	116
1	$\lambda = 0.3$ . . . . .	134
2	$\lambda = 0.5$ . . . . .	134
3	$\lambda = 0.7$ . . . . .	134

# List of Abbreviations

## **Proposed Algorithms**

ASHM Allocation and Split on Heterogeneous Multicore

DM Density Minimization

FDM Frequency Driven Mapping

## **Data-Flow Models**

CSDF Cyclo-Static Data-Flow

SDF Synchronous Data-Flow

## **Mixed-Criticality Systems**

AMC Adaptive Mixed-Criticality

CA Certification Authority

EDF-VD Earliest Deadline First with Virtual Deadline

IMC Imprecise Mixed-Criticality

MC Mixed-Criticality

## **Others**

ASIP Application Specific Instruction Processor

EE Energy-Efficient

ICP Integer Convex Programming

ILP Integer Linear Programming

MPSoC Multi-Processor System-on-Chip

PE Performance-Efficient

UAV Unmanned Aerial Vehicle

VFS Voltage and Frequency Scaling

### **Real-Time Systems**

BF Best-Fit

DBF Demand Bound Function

EDF Earliest Deadline First

FF First-Fit

FFD First-Fit-Decreasing

HRT Hard-Real-Time

LLF Least Laxity First

QPA Quick convergency Processor-demand Analysis

WCET Worst-Case Execution Time

WF Worst-Fit

WFD Worst-Fit-Decreasing