Designing of integrated cyber-phisical systems - course description

General information

Course name	Designing of integrated cyber-phisical systems
Course ID	11.3-WE-INFD-DofIC-PS-Er
Faculty	Faculty of Computer Science, Electrical Engineering and Automatics
Field of study	Computer Science
Education profile	academic
Level of studies	Second-cycle Erasmus programme
Beginning semester	winter term 2021/2022

Course information

Semester	2
ECTS credits to win	5
Course type	obligatory
Teaching language	english
Author of syllabus	• dr hab. inż. Remigiusz Wiśniewski, prof. UZ

Classes forms

The class form	Hours per semester (full-time)	Hours per week (full-time)	Hours per semester (part-time)	Hours per week (part-time)	Form of assignment
Lecture	15	1	-	-	Credit with grade
Laboratory	30	2	-	-	Credit with grade
Project	15	1	-	-	Credit with grade

Aim of the course

- Familiarize students with the designing methods of a control part of an integrated cyber-physical systems.
- Familiarize students with the basic knowledge about specification, analysis, and implementation of a control part of an integrated cyber-physical systems.

Prerequisites

Scope

Introduction: cyber-physical system (CPS), concurrent control system, smart system, Internet of Things (IoT), integrated and distributed systems.

General designing flow of a control part of an integrated cyber-physical system: specification, analysis (validation, verification), decomposition and synchronization, designing of the decomposed system, functional verification, implementation.

Graphical specification methods of a control part of an integrated CPS: Petri nets, interpreted Petri nets, UML diagrams.

Analysis of a control part of an integrated CPS: validation, formal verification, concurrency and squentiality analysis with the application of linear algebra (invariants), graph and hypergraph theories. Liveness, bounedness and safeness. Computational complexity of analysis algorithms.

Decomposition and synchronization of the system: decomposition into sequential automata (state machine components), decomposition methods (linear algebra, graph theory, hypergraph theory), time domains, synchronization of decomposed modules.

Designing of the decomposed system: finite state machines (FSMs), microprogrammed controllers, description in the hardware languages (Verilog, VHDL).

Implementation of the system: logic synthesis, physical implementation of the system in the FPGA device.

Static partial reconfiguration of already implemented system.

Dynamic partial reconfiguration of already implemented system.

Teaching methods

Lecture, laboratory exercises, project.

Learning outcomes and methods of theirs verification

Outcome description	Outcome symbols	Methods of verification	The class form
Has a basic knowledge on designing of a control part of an integrated cyber-physical systems with the application of graphical methods (Petri nets)		• a discussion • activity during the classes • an evaluation test	• Lecture • Laboratory
Has a knowledge about various analysis and decomposition methods of a control part of an integrated cyber-physical system (in regards of computational complexity of algorithms)		• a discussion • activity during the classes • an evaluation test	• Lecture

Outcome description	Outcome symbols	Methods of verification	The class form
Is able to design a control part of an integrated cyber-physical system with the application of graphical specification methods (e.g. Petri nets).		• a project • activity during the classes • an ongoing monitoring during classes	• Laboratory • Project

Assignment conditions

Lecture – the passing condition is to obtain a positive mark from the final test (or other tasks given by the teacher).

Laboratory – the passing condition is to obtain positive marks from all laboratory exercises to be planned during the semester (or other tasks given by the teacher).

Project – the passing condition is to obtain a positive mark from all projects conducted during the semester (or other task given by the teacher).

Final mark components: lecture 30% + laboratory 40% + project 30%.

Recommended reading

1. E. A. Lee, S. A. Seshia, *Introduction to Embedded Systems: A Cyber-Physical Systems Approach*, Cambridge, MA, USA:MIT Press, 2017, https://ptolemy.berkeley.edu/books/leeseshia/releases/LeeSeshia_DigitalV2_2.pdf
2. R. Alur, *Principles of Cyber-Physical Systems*, MIT Press, 2015.
3. W. Reisig, *Petri Nets: An Introduction*, Berlin, Germany:Springer-Verlag, 2012.
4. OMG UML, *Unified Modeling Language*,2012, http://www.omg.org/spec/UML/ISO/19505-2/PDF
5. R. Wiśniewski, *Prototyping of Concurrent Control Systems Implemented in FPGA Devices*, Cham, Switzerland:Springer, 2017, https://link.springer.com/content/pdf/10.1007/978-3-319-45811-3.pdf

Further reading

1. Best, R. Devillers, M. Koutny, *Petri Net Algebra*, Berlin, Germany:Springer-Verlag, 2013.
2. I. Grobelna, R. Wiśniewski, M. Grobelny, M. Wiśniewska, "Design and verification of real-life processes with application of Petri nets", *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 47, no. 11, pp. 2856-2869, Nov. 2017.
3. L. Gomes, A. Costa, J. P. Barros, P. Lima, "From Petri net models to VHDL implementation of digital controllers", *Proc. IEEE 33rd Annu. Conf.*, pp. 94-99, Nov. 2007.
4. R. Wiśniewski, G. Bazydło, L. Gomes, A. Costa, "Dynamic partial reconfiguration of concurrent control systems implemented in FPGA devices", *IEEE Trans. Ind. Informat.* , vol. 13, no. 4, pp. 1734-1741, Aug. 2017.
5. L. Gomes, F. Moutinho, F. Pereira, "IOPT-tools—A Web based tool framework for embedded systems controller development using Petri nets", *Proc. 23rd Int. Conf. Field Program. Logic Appl.*, pp. 1, Sep. 2013.
6. I. Grobelna, "Model checking of reconfigurable FPGA modules specified by Petri nets", *J. Syst. Archit.*, vol. 89, pp. 1-9, Sep. 2018, DOI: http://doi.org/10.1016/j.sysarc.2018.06.005.
7. R. Wiśniewski, "Dynamic partial reconfiguration of concurrent control systems specified by Petri nets and implemented in Xilinx FPGA devices", *IEEE Access*, vol. 6, pp. 32376-32391, 2018, DOI: http://dx.doi.org/10.1109/ACCESS.2018.2836858.
8. M.C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, 1980.
9. R. Wiśniewski, A. Karatkevich, M. Adamski, A. Costa, L. Gomes, "Prototyping of concurrent control systems with application of Petri nets and comparability graphs", *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 2, pp. 575-586, Mar. 2018.
10. R. David, and H. Alla, *Discrete, Continuous, and Hybrid Petri Nets,* Springer, 2005.
11. R. Wiśniewski, G. Bazydło, P. Szcześniak, I. Grobelna, M. Wojnakowski, „Design and Verification of Cyber-Physical Systems Specified by Petri Nets - A Case Study of a Direct Matrix Converter", *Mathematics*, vol. 7, pp. 1-24, 2019, DOI: https://doi.org/10.3390/math7090812.
12. V. Hahanov et al., "Cyber social computing" in Social Business and Industrial Applications, Cham, Switzerland:Springer, pp. 489-515, 2019.
13. M. Szpyrka, M. Wypych, J. Biernacki, L. Podolski, "Discrete-time systems modelling and verification with Alvis language and tools", *IEEE Access*, vol. 6, pp. 78766-78779, Dec. 2018, DOI: https://doi.org/10.1109/ACCESS.2018.2885249.
14. C. Berge, *Hypergraphs: Combinatorics of Finite Sets*, Amsterdam, The Netherlands:North Holland, 1989.
15. R. Wiśniewski, M. Wiśniewska and M. Jarnut, "C-Exact Hypergraphs in Concurrency and Sequentiality Analyses of Cyber-Physical Systems Specified by Safe Petri Nets," *IEEE Access*, vol. 7, pp. 13510-13522, 2019, DOI: https://doi.org/10.1109/ACCESS.2019.2893284.

Notes

Modified by dr hab. inż. Remigiusz Wiśniewski, prof. UZ (last modification: 16-07-2021 22:33)