Discrete process control - course description

General information

Course name	Discrete process control
Course ID	06.0-WE-AutP-DPC-Er
Faculty	Faculty of Computer Science, Electrical Engineering and Automatics
Field of study	Automatic Control and Robotics
Education profile	academic
Level of studies	Erasmus programme
Beginning semester	winter term 2017/2018

Course information

Semester	3
ECTS credits to win	6
Course type	obligatory
Teaching language	english
Author of syllabus	• dr inż. Grzegorz Bazydło

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	30	2	-	-	Exam
Laboratory	30	2	-	-	Credit with grade

Aim of the course

- Familiarize students with discrete control issues in which the control algorithm is implemented as a sequential model (FSM), concurrent model (Petri net, SFC) or hierarchical model (UML state machine).
- Shaping basic skills of modelling control systems and their formal verification.

Prerequisites

- Fundamentals of discrete systems
- Computer architecture

Scope

- Formal specification of discrete process at the behavioral level: flowchart, hierarchical state machine (statechart, UML state machine), hierarchical Petri net.
- Modular behavioral specification of logic control programs using hierarchical SFC and Petri net, definition of SFC, the relationship between SFC and Petri net, modular design, design cores. The role of formal specification in PLC programming.
- UML as a reactive system specification tool. UML state machine diagram, activity diagram, use cases. UML role in documenting and synthesis of software for embedded digital microsystems.
- Formal verification: application of Petri net theory. SAT methods, the use of expert systems.
- Logic controller architecture: microcontroller as a logic controller, digital System-on-Chip (SoC) microsystems. Industrial Programmable Logic Controller (PLC). Embedded, Reconfigurable Logic Controller (RLC).
- Software or structural realization of logic controllers: PLC programming according to IEC1131 based on behavioral specifications. Structural synthesis of embedded controller using formal methods based on behavioral specification. The role of SystemC language and hardware description languages (e.g., VHDL, Verilog) in system synthesis.
- Specification and modeling of binary control algorithms on the system-level using UML and professional CAD systems for digital microsystem design.

Teaching methods

Lecture: conventional lecture.

Laboratory: laboratory exercises.

Learning outcomes and methods of theirs verification

Outcome description	Outcome symbols	Methods of verification	The class form
Student has the ability to create abstract UML models of systems, in which the control is a central element.		• an exam - oral, descriptive, test and other	• Lecture
Student has the knowledge about the formal verification of the controllers.		• an exam - oral, descriptive, test and other	• Lecture
Student has the ability to describe a control program using typical industrial technologies (e.g., PLC).		• an ongoing monitoring during classes	• Laboratory

Outcome description	Outcomesymbols	Methods of verification	The class form
Student has the ability to design PLCs as well as digital circuits.		• an ongoing monitoring during classes	• Laboratory
Student has the knowledge about the classical definition of discrete control, both sequential, concurrent and hierarchical (FSM, Petri net, state machine).		• an exam - oral, descriptive, test and other	• Lecture

Assignment conditions

Lecture: the main condition to get a pass are sufficient marks in written or oral exam.

Laboratory: a condition of pass is to obtain positive grades from all laboratory exercises that are expected to be performed within the laboratory program.

Composition of the final grade: lecture: 50% + laboratory: 50%

Recommended reading

1. Rumbaugh J., Jacobson I., Booch G.: The Unified Modeling Language Reference Manual, Second Edition, Addison-Wesley, USA, 1999.
2. Adamski M., Karatkevich A., Węgrzyn M.: Design of Embedded Control Systems, Springer (USA), New York, 2005.
3. Żurawski R.(Ed.): Embedded Systems Handbook, CRC, Boca Raton, 2006.
4. Harel D., Feldman Y.: Algorithmics: The Spirit of Computing (3rd Edition), Addison-Wesley, USA, 2004.

Further reading

1. Booch G., Rumbaugh J., Jacobson I.: The Unified Modeling Language User Guide, Second Edition, Addison-Wesley, USA, 2005.
2. David D., Alla H.: Petri Nets & Grafcet. Tools for modeling discrete event systems, Prentice Hall, New York, 1992.
3. Gajski D.D, Vahid F., Narayan S., Grong J.: Specification and Design of Embedded Systems, Pretice Hall, Englewood Cliffs, New Jersey, 1994.
4. Jerraya A., Mernet J. (Ed): System-Level Synthesis, Kluwer, Dordecht, 1999.
5. Yakovlev, Gomes L., L. Lavagno (Ed.): Hardware Design and Petri Nets, Kluwers Academic Publishers, Boston, 2000.

Notes

Modified by dr inż. Grzegorz Bazydło (last modification: 10-05-2017 19:56)