Technology
Arts Sciences
TH Köln

Course
STE - Control System Technology

Version: 3 | Last Change: 30.09.2019 14:20 | Draft: 0 | Status: vom verantwortlichen Dozent freigegeben

General information

Long name	Control System Technology
Approving CModule	STE_BaET
Responsible	Prof. Dr. Stefan Kreiser Professor Fakultät IME
Level	Bachelor
Semester in the year	winter semester
Duration	Semester
Hours in self-study	78
ECTS	5
Professors	Prof. Dr. Stefan Kreiser Professor Fakultät IME Kellersohn
Requirements	basic programming skills (procedural language) sampling theorem Boolean algebra discretization of continuous data coding of data finite state machines
Language	German
Separate final exam	Yes

Final exam

Details

Oral examination after written preparation.
Based on a natural language description of a realistic automation task of appropriate complexity, the students develop a suitable model for a concurrent event-discrete control system. They justify the essential structures of their model with reference to typical automation system, development and maintenance requirements as well as task-specific specifications and prove that the model shows the required behaviour and quality, can be implemented on a controller device and can then be used as a control system that solves the given automation task.

Minimum standard

- Students extract the essential relevant information and solution limitations from the task specification and develop a reasonable petri net model of a control system using discrete signals and events of the technical process (signal interpreted petri net), taking into account essential automation quality criteria.
- Students are able to simulate essential model sections in thought experiments and thus prove that the model under consideration meets special, required behavioral elements.
- Students are able to describe and justify an appropriate implementation concept for their specific model on an industrial control device in its essential structures and properties. They show how the individual model elements and structures are mapped to the implementation concept.

Exam Type

Oral examination after written preparation.
Based on a natural language description of a realistic automation task of appropriate complexity, the students develop a suitable model for a concurrent event-discrete control system. They justify the essential structures of their model with reference to typical automation system, development and maintenance requirements as well as task-specific specifications and prove that the model shows the required behaviour and quality, can be implemented on a controller device and can then be used as a control system that solves the given automation task.

Lecture / Exercises

Learning goals

Knowledge

modelling
structure
system borders
system decomposition
system interfaces
system functionality
behavior
state charts (SC)
hybrid nets
concurrency
hierarchy and history
concept of actions
petri nets
place/transition nets (P/T)
net elements
incidence matrix
forward matrix
backward matrix
condition/event nets (C/E)
behavioral analysis
firing sequences
reachability graph

coverability graph (option)

invariants (option)

net properties assessment

liveness

reversibility

boundedness

determinism

signal interpreted petri nets (SIPN)

modeling pattern

complementary place / reservation

arcs

test arc

inhibitor arc

event arc (option)

hierarchy

timed transitions

transition subnets

place subnets

concept of pages

calculated arc weight

---

control system

signal processing

realtime

types

sources of time conditions

discretization

value axis

time axis

sensors

structure of sensor systems with respect to signal processing

calibration (option)

actuators

structure of actuator systems with respect to signal processing

controller devices

IPC

program organization

resources

RTOS

tasks and threads

scheduling

device categories

µC-Boards

process computer

PAC

RTU

PLC

EN61131

configuration

resources

cyclic tasks

IO variables

program organization

POU

data types

function blocks
programming languages
overview
procedural languages (ST)
graphical languages (FB)
pattern driven realization of SIPN on PLC
examples of controller devices
distributed automation systems
communication
structures
star
bus
ring
redundancy
methods
shared memory
message passing
asynchronous
synchronous
rendezvous
futures
OSI model
protocol layers
MAC
deterministic
non deterministic
field busses
industrial (EN61158)
Interbus
Profibus
Profinet
automotive (option)
CAN
Flexray
area networks
protocol layers
IEEE802
IP
transport protocols
UDP
TCP
SCTP
Industrial Ethernet
hardware
QoS (option)
redundancy (RSTP)
virtual nets (VLAN)
process control systems (PCS, SCADA systems)
EN 61499
architecture
programming
safety
device related safety
network related safety
MES and ERP (option)
object tracking (option)

automatic object identification (AutoID)

object history

protocols

Skills

modeling event driven systems (behavior)

derive system behaviour from comprehensive technical documents

capture any essential information out of technical documents

recognize implicit information

identify and resolve missing information

model as state chart

recognize finite state chart (FSC) as special form

signal interpreted net (SIN)

model as petri net

CE net

PT net

know syntax

consistently and constructively use pattern and makros

hierarchical nets

use deep hierarchy

use flat hierarchy

signal interpreted net (SIN)

consistently and constructively use petri net development tools

verify models

define suitable criteria

equivalence

completeness

determinism

liveness

reversibility

boundedness

meet given modeling assumptions

...

define test cases

conduct model reviews

by own

with peer

graphical analysis

mathematical analysis

conduct dynamic tests using model simulator

correct and optimize models based on review and test results

---

control system design

real time aspects

derive real time conditions

choose control devices

choose bus systems

show real time capabilities of control systems

programming PLC with ST (EN61131-3)

use ST programming syntax

use function blocks

model driven development

design coding templates (pattern) to transform SIPN models into PLC

design code generator to transform SIPN models into PLC
based on C/E nets
based on P/T nets
modeling control flows in a PCS (EN61499)

Expenditure classroom teaching

Type	Attendance (h/Wk.)
Lecture	2
Exercises (whole course)	1
Exercises (shared course)	0
Tutorial (voluntary)	1

Separate exam

none

Project

Learning goals

Skills

programming control systems
consistently and constructively use professional PLC-IDE
configure essential attributes of a PLC device
consistently use ST programming language
use synchronous message passing
constructively use function blocks in programs

---

use target simulator in interaction with PLC IDE

---

manage complex tasks as a team
plan and control small projects
meet agreements and deadlines
plan and conduct reviews

---

modelling real world systems
system analysis
derive system structure and system behaviour from comprehensive technical documents
evaluate and take account of system borders and system interfaces

decompose system structure

define useful subsystems

define subsystem functions

define subsystem interfaces

develop controller model

design hierarchical controller model

model controller subsystems as SIPN

verify and evaluate controller subsystem models

conduct dynamic test using petri net simulator

conduct peer review

integrate controller subsystem models

verify and evaluate controller model using petri net simulator

---

program PLC controller

configure PLC

define cyclic tasks

use given IO-variables

use given user interface

use model transformations

transform controller subsystem models into ST programs using transformation pattern

integrate controller subsystem programs on PLC

verify controller program on PLC

test subsystems using target system emulator

conduct integration test using target system emulator

---

launch controller on target system

Expenditure classroom teaching

Type	Attendance (h/Wk.)
Project	1
Tutorial (voluntary)	0

Separate exam

Exam Type

working on projects assignment with your team e.g. in a lab)

Details

attendance phase with 3 times of 4h of presence per project group, final presentation

Minimum standard

Finding suitable system boundaries and modelling a hierarchical control system and the planned subsystems.

Control system implementation on a professional controller device.