

Lehrveranstaltung der Regensburg School of Digital Sciences (RSDS)

(Modul-)Titel	Falls vorhanden Modulbez. oder -nr.	
Prozessinformatik (Process Computer Science)	RSDS-PI	
(Modul-)Verantwortliche/r	Fakultät	
Prof. Dr. Wolfgang Bock	Maschinenbau	
Lehrende/r / Dozierende/r	Angebotsfrequenz	
Prof. Dr. Wolfgang Bock	jedes Semester	
Lehrform	Unterrichtssprache	
Seminaristischer Unterricht	deutsch	
Art der Prüfung	Voraussetzungen	
Schriftliche Prüfung, 90 Min.	Grundlagen der Programmierung	
Teilnehmerzahl (gesamt)	Modultyp	Arbeitsaufwand
20		4 SWS / 5 ECTS
Zielfakultäten/ -studiengänge (inkl. Teilnehmerzahl pro Studiengang)	Für Bachelor	Für Master
PA ISE ME Geöffnet für Studierende aus: EI, REE MB	✓	✗
Inhalt (Kurzbeschreibung)		
<ul style="list-style-type: none"> • Automatisierungssysteme: Begriffsbestimmung, Grundfunktionen • Fundierte Kenntnisse zu den Grundbegriffen und Normen der Industrieautomation • Hard- und Softwaremodell der IEC 61131, Normen und Vorgehensweisen für eine systematische Software- Entwicklung • Beschreibung von Steuerungsalgorithmen mit UML- Methoden, insbesondere OOP und Graphen • Programmiersprachen: Strukturierter Text, Anweisungsliste, Funktionsplan, Ablaufsprache, objekt-orientierte Sprachelemente • Einfache, zusammengesetzte und spezielle SPS-Datentypen • Vertiefte theoretische und praktische Kenntnisse zur Codierung von Prozessabläufen • Integrierte Entwicklungsumgebungen: Konfiguration und Parametrierung • Programmiertechniken: Strukturierte Programmierung, Schrittkettenprogrammierung, SPS-Hochsprachen, Zustandsautomaten • Organisation von Softwareprojekten: Strukturierung, Bibliotheken, Wiederverwendbarkeit 		

- Prozessvisualisierung: Grundbegriffe und Übungen
- Buskommunikation in der Industrieautomation: allgemeine Grundlagen und konkrete Beispiele
- ISO/OSI-Kommunikationsmodells am Beispiel von TCP/IP und weiteren Bussystemen der Prozessinformatik

Lernziel

Nach der erfolgreichen Absolvierung des Teilmoduls sind die Studierenden in der Lage,

Fachkompetenz:

- ein steuerungstechnisches Softwareprojekt zu entwerfen (3) und die dazu passenden Programmorganisationseinheiten (POEen) zu erstellen (3)
- an eine Automatisierungsaufgabe methodisch heranzugehen (3)
- eine zugrundeliegende Logikfunktion zu finden, zu minimieren und mit Schaltnetzen zu programmieren (3)
- Schaltwerke unter Verwendung von Flipflops, Timern und Countern anzulegen und zu parametrieren (2)
- logische, arithmetische und programmverzweigende Anweisungen zur Modellierung von Prozessabläufen zu formulieren (2)
- mit aktuellen SPS-Entwicklungsumgebungen Projekte zu codieren, speichern, simulieren und debuggen (2)
- Struktogramme für Algorithmen zu erstellen und diese in der Sprache Strukturierter Text umzusetzen (2)
- Ablauf- und Zustandsgraphen zu erstellen (2) und durch Codierung, Verifikation und Simulation umzusetzen (3)

Persönliche Kompetenz:

- Programmieraufgaben in kleinen Teams zu lösen (2)
- mit Fachbegriffen in deutscher und englischer Sprache umzugehen (1)
- die Auswirkungen der Automatisierungstechnik auf die Arbeitswelt abzuwägen (3)
- Gefährdungen des Menschen durch automatisierte Prozessabläufe zu beurteilen (3)

Die Zahlen in Klammern geben die zu erreichenden Niveaustufen an: 1 - kennen, 2 - können, 3 - verstehen und anwenden