### Module INF-EXP-951: Cyber-Physical System Fundamentals (CPSF)

#### Module Structure

<table>
<thead>
<tr>
<th>No</th>
<th>Module / Course</th>
<th>Type</th>
<th>Credits</th>
<th>SWS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cyber-Physical System Fundamentals</td>
<td>V</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>Cyber-Physical System Fundamentals Lab</td>
<td>P</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

#### Language: english

#### Content

The course is based on the presenter’s book on the subject and includes the following topics:

1. Introduction: Definition of terms, scope of the course
2. Specification and modeling: models of computation, communication models, finite state machines, data flow, discrete event models, von-Neumann-models, expressiveness of models
3. CPS hardware: hardware-in-the-loop, sampling and A/D-conversion, processing, field-programmable gate arrays (FPGAs), communication hardware, D/A-conversion, sampling theorem, output
4. Standard software: embedded operation systems, real-time operation systems, priority inversion, middleware
5. Evaluation and validation: objective functions, Pareto-optimality, worst-case execution time, energy consumption, reliability, real-time calculus, verification
6. Mapping of applications to execution platforms: standard optimization techniques, real-time scheduling, rate monotonic scheduling, earliest deadline first scheduling, hardware/software partitioning, mapping of applications to heterogeneous multiprocessors
7. Selected optimizations

#### Literature

- Peter Marwedel: Embedded System Design – Cyber Physical System Fundamentals, Springer 2010
- Lego Mindstorm NTX technical documentation
- Technical documentation for the used finite state machine design tool (StateMate or similar)

#### Goals

Students successfully finishing the course should be able to
- understand how cyber-physical (CPS) hardware interacts with CPS software and use this knowledge to design CPS software,
- select models of computation and programming languages that are appropriate for a given design problem,
- select an appropriate scheduling technique for embedded systems,
- apply hardware/software codesign techniques in order to optimize the system which they are supposed to design.

#### Examinations

**Module examination**: written examination

**Course achievement:**
- successful completion of element 2

The course achievement is a prerequisite for the module examination.

#### Type of Examination

- [x] Module Examination
- [ ] Cumulative Examinations

#### Requirements†

- none if attended as a master’s degree course –

---

*Bitte beachten Sie, dass die Leistungspunkte je nach Prüfungsordnung abweichen können.
† Bitte beachten Sie, dass die Teilnahmevoraussetzungen je nach Prüfungsordnung abweichen können.
<table>
<thead>
<tr>
<th>8</th>
<th>Module Type and Allocation to Curriculum</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>see regulations for the resp. degree</td>
</tr>
<tr>
<td></td>
<td>Students can either obtain credit points for this module or INF-BSc-232 „Eingebettete Systeme (ES)“, but not for both.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>9</th>
<th>Responsible</th>
<th>Department</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Prof. Dr. J.-J. Chen</td>
<td>Computer Science</td>
</tr>
</tbody>
</table>