

**Module Description, available in: EN, FR**

## *Design of Embedded Hardware and Firmware*

**General Information****Number of ECTS Credits**

3

**Module code**

TSM\_EmbHardw

**Valid for academic year**

2023-24

**Last modification**

2019-07-22

**Coordinator of the module**

Hans Dermot Doran (ZHAW, hans.doran@zhaw.ch)

**Explanations regarding the language definitions for each location:**

- Instruction is given in the language defined below for each location/each time the module is held.
- Documentation is available in the languages defined below. Where documents are in several languages, the percentage distribution is shown (100% = all the documentation).
- The examination is available 100% in the languages shown for each location/each time it is held.

	Lausanne		Lugano	Zurich	
<b>Instruction</b>		X F 100%		X E 100%	
<b>Documentation</b>			X E 100%	X E 100%	
<b>Examination</b>		X F 100%		X E 100%	

**Module Category**

TSM Technical scientific module

**Lessons**

2 lecture periods and 1 tutorial period per week

**Entry level competences****Prerequisites, previous knowledge**

The students have a working knowledge of programming embedded systems in C.  
The students have a working knowledge of basic hardware design including VHDL coding

**Brief course description of module objectives and content**

This module introduces the student to advanced concepts in modern embedded systems engineering. The module is divided into two sections. The first section is practical/theoretical and is designed to get the student familiar with implementing System on Chip (SoC) designs. The second part discusses formal Hardware/Software Co-Design including design and implementation of advanced embedded architectures as well as the verification and test of the resulting system.

## Aims, content, methods

### Learning objectives and acquired competencies

- The student will know some of the forces driving the design of modern embedded architectures.
- The student will understand and be able to apply the V-Model and structured HW/SW Co-Design methodologies including strategies for the verification and test of embedded systems.
- The student will be able to design and implement complete SoC designs including using soft-core microprocessors and IP cores in an FPGA.
- The student will be able to apply loop optimisations using both SW techniques and optimised cache in single- and multi-processor architectures.
- The students will be able to understand and apply pipeline architectures in processors (super-pipelined, superscalar), HW and SW.

### Contents of module with emphasis on teaching content

- Introduction
  - V-Model, specification and test
  - HW-SW Co-Design
- SoC design, implementation and test
  - FPGA technology, SoC design, soft-core processors, design, implementation and reuse of custom IP cores
  - Bus systems
- Optimisation Strategies
  - Advanced peripherals, DMA, scheduling
  - Software loop optimisations, custom instructions, co-processors,
  - Memory hierarchy (cache, scratch pad memories )
  - Pipeline, multiprocessing
- Review
  - Exercises and laboratories using an FPGA board

### Teaching and learning methods

Lectures  
Accompanied exercises  
Self-study

### Literature

No mandatory literature

## Assessment

### Certification requirements

Module uses certification requirements

### Certification requirements for final examinations (conditions for attestation)

Part 1: Voluntary test or presentation of Practical Work - SoC Implementation (20%)

Part 2: Voluntary test or presentation of Practical Work - Formal Processes and Methods (20%)

### Basic principle for exams

**As a rule, all the standard final exams for modules and also all resit exams are to be in written form**

### Standard final exam for a module and written resit exam

#### Kind of exam

written

#### Duration of exam

120 minutes

#### Permissible aids

*Aids permitted as specified below:*

Permissible electronic aids

No electronic aids permitted

**Other permissible aids**

Lecture notes and VHDL reference

**Special case: Resit exam as oral exam**

**Kind of exam**

oral

**Duration of exam**

30 minutes

**Permissible aids**

*Aids permitted as specified below:*

**Permissible electronic aids**

No electronic aids permitted

**Other permissible aids**

Lecture notes and VHDL reference

Description du module, disponible en: EN, FR

## Conception de hardware et firmware embarqués

### Informations générales

Nombre de crédits ECTS

3

Code du module

TSM\_EmbHardw

Valable pour l'année académique

2023-24

Dernière modification

2019-07-22

Coordinateur/coordinatrice du module

Hans Dermot Doran (ZHAW, hans.doran@zhaw.ch)

Explication des définitions de langue par lieu :

- Les cours se dérouleront dans la langue définie ci-dessous par lieu/exécution.
- Les documents sont disponibles dans les langues définies ci-dessous. Pour le multilinguisme, voir la répartition en pourcentage (100% = documents complets)
- L'examen est disponible à 100% dans chaque langue sélectionnée pour chaque lieu/exécution.

	Lausanne		Lugano	Zurich	
<b>Leçons</b>		X F 100%		X E 100%	
<b>Documentation</b>			X E 100%	X E 100%	
<b>Examen</b>		X F 100%		X E 100%	

Catégorie de module

TSM approfondissement technico-scientifique

Leçons

2 leçons et 1 leçon de pratique par semaine

### Compétences préalables

Connaissances préalables, compétences initiales

Les étudiants jouissent de connaissances professionnelles des systèmes de programmation en C.

Les étudiants jouissent de connaissances professionnelles des bases de la conception matérielle y compris le codage VHDL

### Brève description du contenu et des objectifs

Le présent module donne aux étudiants un aperçu sur les concepts avancés de technologies modernes d'ingénierie embarquée. Le module se subdivise en deux parties. La première partie est consacrée à la pratique/théorie et vise à familiariser l'étudiant à mettre en oeuvre des systèmes de conception de puces. La deuxième partie aborde la conception conjointe formelle matérielle et logicielle, y compris la conception et la mise en oeuvre d'architectures embarquées et la vérification des systèmes créés.

## Objectifs, contenus, méthodes

### Objectifs d'apprentissage, compétences à acquérir

- L'étudiant connaîtra quelques-unes des forces intervenant dans la direction des architectures embarquées modernes.
- L'étudiant comprendra et sera capable d'appliquer des méthodes de co-design matériel et logiciel, ainsi que des stratégies de test et de vérification des systèmes embarqués (vérification conjointe HW/SW).
- L'étudiant sera capable de concevoir et de mandater des designs SoC complet sur un FPGA.
- L'étudiant sera capable de concevoir des SoC au moyen des technologies suivantes- processeurs softcore (multiples), co-processeurs (instructions programmables, coprocesseurs fortement couplés, processeurs de signal) et accélération matérielle.
- L'étudiant comprendra et appliquera les techniques d'optimisation logicielle avancées.
- L'étudiant devra réaliser des exercices pertinents sur une carte de développement adéquate.

### Contenu des modules avec pondération du contenu des cours

- Introduction
  - Technologie FPGA
  - conception SoC, processeurs soft-core, modules auto-conçus
  - systèmes de bus, DMA, hiérarchie mémoire (caches, SPM)
- Architectures des automates
  - Processeurs softcore, instructions programmables, coprocesseurs, architectures de processeurs
  - Firm/software embarqués et techniques d'optimisation
  - Interfaçage périphérique, accélération matérielle
- Test et vérification
  - Vérification conjointe matérielle et logicielle et stratégies de contrôle
- Examen
  - Exercices et laboratoire utilisant une carte FPGA

### Méthodes d'enseignement et d'apprentissage

Cours magistraux  
Travaux dirigés  
Etude autonome

### Bibliographie

Pas de bibliographie obligatoire

## Evaluation

### Conditions d'admission

Le module utilise les conditions d'admission

### Conditions d'admission à l'examen de fin de module (exigences du certificat)

Test volontaire partie 1: Implémentation SoC (20%)  
Test volontaire partie 2: Processus et méthodologie formels (20%)

### Principe pour les examens

**En règle générale, tous les examens de fin de module réguliers et les examens de rattrapage sont organisés sous la forme écrite**

### Examen de fin de module régulier et examen écrit de répétition

#### Type de l'examen

écrit

#### Durée de l'examen

120 minutes

#### Aides autorisées

*Les aides suivantes sont autorisées:*

#### Aides électroniques autorisées

Aucune aide électronique autorisée

#### Autres aides autorisées

**Cas spécial: examen de répétition oral**

**Type de l'examen**

oral

**Durée de l'examen**

30 minutes

**Aides autorisées**

*Les aides suivantes sont autorisées:*

**Aides électroniques autorisées**

Aucune aide électronique autorisée

**Autres aides**

Lecture notes and VHDL reference