Programmation Embarquée
Programming embedded systems

Code ECUE *Course code:* **PEM**

UE (Crédits ECTS de l'UE) : UE4-2c (9 ECTS)

Département <i>Department</i>	IA	Cours <i>Lectures</i>	07h30
Coordonnateurs <i>Lecturers</i>	E. Grolleau	T.D. <i>Tutorials</i>	07h30
Période <i>Year of study</i>	A2 2 nd year	T.P. <i>Laboratory sessions</i>	03h00
Semestre <i>Semester</i>	S4 4 th semester	Projet <i>Project</i>	
Evaluation <i>Assessment method(s)</i>	1 écrit <i>1 written exam</i>	Non encadré <i>Unsupervised</i>	
Langue d'instruction <i>Language of instruction</i>	Français <i>French</i>	Horaire global <i>Total hours</i>	18h00
Type de cours <i>Type of course</i>	Majeure (S4) <i>Major (4th semester)</i>	Travail personnel <i>Homework</i>	07h00
Niveau <i>Level of course</i>	Second cycle universitaire <i>Graduate</i>		

Compétences attendues :

- Détecter les problèmes potentiels lors du ciblage des programmes embarqués sur différentes architectures matérielles, nécessite les connaissances de :
 - Grandes familles de calculateurs et leurs différences (32 vs 64 bits, utilisation de la pile d'appel, FPU, architectures vectorielles) ;
 - L'impact de l'architecture sur la représentation mémoire (tailles de représentation entiers, alignement sur les mots mémoire, etc.)
 - L'impact des optimisations locales sur la durée d'exécution (mémoires caches, RAM, pipeline, prefetch, etc.)
 - Connaître les spécificités des processeurs multicœurs et pluricœurs
- Connaissance du fonctionnement bas niveau des programmes sur différentes architectures (8, 32, 64 bits notamment pour architectures ARM, x86, etc.)

Pré-requis :

- Informatique Numérique (S1)
- Introduction aux Systèmes Embarqués (S1)
- Ingénierie des Systèmes critiques (S3)

Contenu :

- Architectures matérielles
 - Le goulot d'étranglement de la mémoire des architectures Harvard/von Neumann à nos jours (banques SDRAM) et mémoires cache
 - Le partitionnement mémoire par mémoire virtuelle
 - Structure interne d'un processeur
- Architectures parallèles
 - Hyper-threading
 - Multicoeur
 - Architecture MPSoC hétérogènes
 - Pluricoeur
- Architectures logicielles
 - RTOS vs GPOS
- Code bas niveau
 - Compilation vers code assembleur et code machine
 - Fonctionnement de la pile et du tas

- Différences architectures 32 bits et 64 bits
- Option de compilation et optimisation du code produit

Bibliographie :

- E. Grolleau, J. Hugues, Y. Ouhammou, H. Bauer, « Introduction aux systèmes embarqués temps réel, Conception et mise en œuvre », Dunod, 2018
 - F. Cottet, E. Grolleau, S. Gérard, J. Hugues, Y. Ouhammou, S. Tucci-Piergiovanni, « Systèmes temps réel embarqués - 2e édition, Spécification, conception, implémentation et validation temporelle », Dunod, 2014
-

Expected competencies:

- Detect problems when porting a program on an embedded target
 - Major computer families and their differences (32 vs. 64 bits, use of call stack, FPU, vector architectures) ;
 - The impact of architecture on memory representation (integer representation sizes, alignment on memory words, etc.).
 - The impact of local optimizations on runtime (caches, RAM, pipeline, prefetch, etc.).
 - Knowledge of the specific features of multicore and multi-core processors.
- Knowledge of low-level program operation on different architectures (8, 32, 64-bit, especially for ARM, x86 architectures, etc.).

Prerequisites: Digital Information (S1), Introduction to Embedded Systems (S1), Model-based system engineering for critical systems (S3)

Content:

- Hardware architectures
- Parallel architectures (Hyper-threading, multicore, heterogeneous MPSoCs, manycore)
- Software architectures: RTOS vs GPOS
- Low-level code

Recommended reading:

- E. Grolleau, J. Hugues, Y. Ouhammou, H. Bauer, « Introduction aux systèmes embarqués temps réel, Conception et mise en œuvre », Dunod, 2018
- F. Cottet, E. Grolleau, S. Gérard, J. Hugues, Y. Ouhammou, S. Tucci-Piergiovanni, « Systèmes temps réel embarqués - 2e édition, Spécification, conception, implémentation et validation temporelle », Dunod, 2014