



---

**Systèmes Embarqués 1 & 2**  
**Objectifs du travail écrit n° 3**

**Les étudiant-e-s devront être capable :**

**Interfaçage assembleur - C**

- d'utiliser correctement la pile (stack) dans un programme codé en assembleur
- de programmer une sous-routine en assembleur ARM en utilisant la pile pour le passage de paramètres et pour les variables locales
- de représenter le passage de paramètres à des sous-routines et d'expliquer le retour de valeurs et du résultat pour interfacé avec des programmes écrits en C
- de représenter l'image de la pile lors de l'appel d'une sous-routine en assembleur
- de coder dans les règles de l'art l'appel de routines développées en C depuis un code en assembleur ARM
- de spécifier correctement l'appel de routines en assembleur depuis un code C
- d'expliquer les conditions pour qu'une sous-routine soit réentrante

**Interruptions**

- d'expliquer les différentes phases de traitement d'une interruption (séquence)
- de classer les types d'interruptions et d'exceptions
- d'expliquer la fonctionnalité de la table des vecteurs d'interruptions sur les processeurs ARM, ainsi que l'implémentation spécifique du processeur AM335x
- d'expliquer comment l'on peut passer du mode superviseur au mode utilisateur et vice versa
- d'identifier l'état du processeur et de la pile lors du traitement d'une interruption
- d'utiliser correctement les pointeurs de pile dans les différents modes des processeurs ARM
- de décrire le concept de commutation de contexte, de latence et de gigue
- de décrire le système d'interruption des processeurs ARM et du processeur AM335x
- d'expliquer les mécanismes d'activation et de désactivation des interruptions matérielles (processeur ARM – core –, processeur AM335x – intc, gpio, ... –)
- de décrire le principe de niveaux de priorité
- de décrire la procédure de reconnaissance d'interruption multiple
- de décrire et de concevoir un gestionnaire d'interruption
- de programmer en assembleur une application utilisant une interruption
- de décrire et de concevoir des opérations atomiques



**Systèmes Embarqués 1 & 2**  
**Objectifs du travail écrit n° 3**

**Systèmes d'exploitation**

- de citer quelques techniques et méthodes de développement pour des systèmes embarqués
- de décrire les différents types de systèmes multitâches (systèmes d'exploitation)
- de décrire les différentes composantes d'un noyau
- de décrire les ressources globales, partagées et privées d'un thread
- de décrire les états principaux d'un thread
- de décrire les éléments et structures nécessaires à la gestion d'un thread
- de décrire la commutation de contexte entre deux threads
- de concevoir un algorithme de transfert de contexte entre deux threads
- de concevoir l'initialisation du contexte d'un thread
- de concevoir un ordonnanceur (scheduler) élémentaire
- de concevoir un mécanisme de synchronisation simple (sémaphore)

Le seul document à disposition est le « 05\_ARM\_Instruction\_Set\_Summary.pdf »  
(jeu d'instructions du processeur ARM)