

# X5I0030: architecture des ordinateurs

Syllabus 2013/2014

## Description du cours

Le cours présente l'architecture des ordinateurs à bas niveau; les conséquences sur le développement logiciel à haut niveau sont évoquées (présence de mémoire cache, limitation des modes d'adressage supportés). On partira de la représentation interne de l'information en binaire (en présentant les différents codages pour les nombres entiers signés et le codage IEEE 754 pour les nombres flottants), puis on montrera comment une instruction machine est décodée par un processeur de type MIPS pour être exécutée (chemin de données). La construction d'un processeur simple monocycle à partir de briques de base (logique combinatoire et séquentielle) sera décrite en détails. L'apprentissage de l'assembleur MIPS permettra d'appréhender la notion de registre et d'adressage de la mémoire. En point subsidiaire, effectué selon l'état d'avancement du cours en fin de semestre, les technologies utilisées pour les différents périphériques (entrée/sortie ou supports de masse) seront présentées, avec un accent sur leur impact dans le cadre de l'interaction de ces périphériques avec le processeur et la mémoire.

## Organisation

Le cours est composé de 12h de cours magistraux présentant les points importants du cours à l'aide de transparents (un travail personnel d'approfondissement des notions évoquées est attendu — voir les références ci-dessous); de 20 heures de travaux dirigés au cours desquels seront faits des exercices permettant de valider la compréhension des notions vues en cours; et de 16 heures de travaux pratiques devant un ordinateur. Lors des travaux pratiques, on implémentera en C ou en C++ certains algorithmes de décodage de l'information binaire, on utilisera le logiciel `logisim` pour créer des circuits combinatoires et séquentiels (additionneur, `UaL`, ...) et le logiciel `MARS` pour l'exécution de code assembleur MIPS sur une machine virtuelle.

## Objectifs du cours

- Connaître la représentation des informations en binaire en machine (entiers signés/non signés, nombres flottants, caractères et chaînes de caractères, instructions);
- Comprendre l'organisation des processeurs et les mécanismes de communication avec les périphériques via les bus (mémoire, disques durs, ...);

- Savoir construire un processeur à partir d'éléments logiques simples (portes ET, OU, ...);
- Comprendre l'impact de l'architecture d'un ordinateur sur les performances d'un programme écrit dans un langage de haut niveau.

## Compétences acquises

- Connaissance de la représentation bas niveau des données manipulées par un programme;
- Connaissance détaillée de l'architecture d'un processeur et de sa décomposition en unités logiques;
- Connaissance pratique de l'assembleur MIPS;
- Connaissance des technologies utilisées pour stocker des données (en mémoire ou sur support de masse).

## Notation

La note globale du module se décompose de la façon suivante :

Contrôle continu	contrôle « sur table »	30%	50%
	Exercice de TP	5%	
	Projet de TP	15%	
Examen		50%	

- Le contrôle continu sur table se fait dans les conditions de l'examen et porte sur tout le cours vu auparavant;
- L'exercice de TP correspond à une séance de travaux pratiques notée portant sur la programmation en assembleur MIPS;
- Le projet de TP est un projet de fin de module correspondant généralement en l'implémentation dans `logisim` d'un processeur simple ou d'une partie de processeur. Le projet se déroule sur 6 heures de travaux pratiques encadrés, les étudiants étant supposés travailler de façon autonome entre chaque séance.