

# Outils de CAO pour VLSI

Présentation du flot de conception VLSI Alliance

Suite du module VLSI de M1 ou MVLSI de M2

Franck Wajsbürt / Sophie Belloeil (UPMC) / Stéphanie Youssef (UFE)

- 6 cours de 2 heures
  - ◆ Flot de conception => 3 vues / vhdl / automate
  - ◆ Synthèse logique => automatique et manuelle
  - ◆ Synthèse physique => génération du layout symbolique et réel
- 4 TME de 4 heures
  - ◆ Synthèse logique => du comportement à la netlist
  - ◆ Langage Stratus => description netlist
  - ◆ Placement routage => de la netlist au layout
  - ◆ Vérification => test de circuit et passage au réel

Cet UE fait suite aux modules VLSI de M1 ou MVLSI de M2. Il est suivi par les étudiants de l'UPMC et de l'UFE.

Il s'organise en deux parties: d'abord 3 séances de 2 fois 2 heures de cours. puis 4 séances de 4 heures de TME qui seront assurées par Sophie Belloeil à Paris et Stéphanie Youssef au Caire.

La première séance de cours présente le flot de conception d'un VLSI qui permet de passer de la description d'un modèle comportementale aux masques attendus par le fondeur, puis approfondit la modélisation des automates. La seconde séance porte sur les outils de synthèse, c'est-à-dire ceux qui permettent de passer de la description d'un comportement à une interconnexion de portes logiques. Nous verrons en particulier le langage Stratus bâti sur Python. La troisième séance s'intéresse à la synthèse physique c'est-à-dire le passage d'une netlist aux masques, au dessin de cellules et aux outils de test.

Les 2 premières séances de TME portent sur les outils et les langages de modélisation, le passage de la vue comportementale à la vue netlist. Les 2 suivantes sur les outils et les langages qui permettent le passage de la vue netlist à la vue physique.

L'évaluation se base sur les comptes rendus de TP et un examen dans lequel vous devez décrire un circuit et décrire un Makefile.

## Cours

- Flot de conception / Modélisation
- Automates synchrones
- Synthèse logique
- Langage Stratus / Python

## TME

- TP1 : Synthèse logique

# Annales