MPRI 2.23-1: Synchronous Parallelism

http://www.di.ens.fr/~pouzet/cours/mpri/

- Teaching: 24h (8 lessons) + 2 TPs; 3 ECTS
- Beginning: 12/09/17, 12h45 – 15h45, Univ. Paris 7, Bat. Sophie Germain.
- Évaluation:
  - Exam (3h) + programming project or presentation of a research paper.
  - Final mark = 1/2 Exam + 1/2 project
- Teachers for 2017.

Marc Pouzet	Timothy Bourke	Jean Vuillemin
		
Prof. P6/ENS	CR INRIA	Prof. ENS

http://www.di.ens.fr/ParkasTeam.html

Scientific Objective of this Course

Learn the foundamental concepts of **synchronous parallelism** and some of the major **synchronous programming languages**.

Their design, semantics, implementation, formal verification and mathematical models to reason about embedded systems.

Application domain:

- Control/command of **critical embedded systems**: fly-by-wire command, braking systems, train tracking and on-board control.
- People have invented **Domain specific languages**;
- both expressive and offering strong **safety and efficiency guaranties** at compile-time.

Synchronous parallelism is a major trend in embedded system design

More fundamentaly:

- How to model time in a program?
- What are the causalities (cause/consequence) between concurrent/parallel computation? Is a program deterministic?

Technical content of the course

6 courses: Foundations, languages and verification (LM and MP)

- Introduction. Examples and synchronous languages. Principles.
- Semantics and compilation to software and hardware.
- Relaxing synchrony: quasi-synchrony, N-synchrony.
- Synchronous observers and model checking of safety properties.
- Mix of discrete and continuous time for hybrid systems.
- Formalisation of a Lustre compiler to C code.

2 TPs (TB and MP)

- Programming with Lustre; a small compiler for MiniLustre

2 courses: boolean functions representations (JV)

- combinatorial logic and latches; programmable circuits (FPGA)
- BDDs and z-BDD