### Advanced computer architecture

**Ienne Paolo**

<table>
<thead>
<tr>
<th>Cursus</th>
<th>Sem.</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer science minor</td>
<td>E</td>
<td>Opt.</td>
</tr>
<tr>
<td>Computer science</td>
<td>MA2, MA4</td>
<td>Obl.</td>
</tr>
<tr>
<td>Cyber security minor</td>
<td>E</td>
<td>Opt.</td>
</tr>
<tr>
<td>Cybersecurity</td>
<td>MA2, MA4</td>
<td>Obl.</td>
</tr>
<tr>
<td>Electrical and Electronical Engineering</td>
<td>MA2, MA4</td>
<td>Opt.</td>
</tr>
<tr>
<td>SC master EPFL</td>
<td>MA2, MA4</td>
<td>Opt.</td>
</tr>
</tbody>
</table>

**Contact language**: English  
**Credits**: 8  
**Session**: Summer  
**Semester**: Spring  
**Exam**: Written  
**Workload**: 240h  
**Weeks**: 14  
**Hours**: 5 weekly  
**Lecture**: 3 weekly  
**Project**: 2 weekly  

**Number of positions**:  

---

**Summary**

The course studies techniques to exploit Instruction-Level Parallelism (ILP) statically and dynamically. It also addresses some aspects of the design of domain-specific accelerators. Finally, it explores security challenges based on microarchitectural features and hardware isolation techniques.

**Content**

*Pushing processor performance to its limits:*
- Principles of Instruction Level Parallelism (ILP)
- Register renaming techniques
- Prediction and speculation
- Simultaneous multithreading
- VLIW and compiler techniques for ILP
- Dynamic binary translation

*Domain specific architectures and accelerators:*
- Specificities of embedded vs. general computing processors
- Overview of DSPs and related compilation challenges
- High-Level Synthesis and accelerators

*Hardware security:*
- Information leakage through the microarchitecture
- Trusted Execution Environments
- Physical side-channel attacks

**Keywords**


**Learning Prerequisites**

Required courses
• CS-208 Computer Architecture I

Recommended courses
• CS-209 Computer Architecture II

Important concepts to start the course
Undergraduate knowledge of digital circuit design and of computer architecture

Learning Outcomes
By the end of the course, the student must be able to:
• Design strategies to exploit instruction level parallelism in processors.
• Contrast static and dynamic techniques for instruction level parallelism.
• Design effective processor (micro-)architectures for which efficient compilers can be written.
• Develop hardware accelerators competitive to best commercial processors
• Defend against security threats based on microarchitectural processor features

Teaching methods
Courses, labs, and compulsory homeworks.

Assessment methods
Homeworks (30%)
Final exam (70%)

Supervision
Office hours No
Assistants Yes
Forum Yes

Resources
Virtual desktop infrastructure (VDI)
No

Bibliography

Ressources en bibliothèque
• Computer Architecture / Hennessy

Moodle Link
• https://go.epfl.ch/CS-470

Prerequisite for
• CS-471 Advanced Multiprocessor Architecture