

 université PARIS-SACLAY <hr/> IUT DE CACHAN	<p>Université Paris-Saclay — IUT de CACHAN</p> <p>BUT-GEii</p> <p>Année 2024/2025 — semestre S3</p> <p>Développement en environnement libre pour CPU RISC-V</p> <p>Cahier des Charges de la formation</p> <p>Septembre 2024</p>	<p>CdC</p> <p>BUT-GEii</p> <p>RISC-V</p> <p>J.O.Klein</p>
---	---	--

En réponse à la demande de la *Direction Nationale de la Souveraineté Numérique (DNSN)*, le département GEii-1 de l'IUT de Cachan propose une formation sur 4 demi-journées, caractérisée par les objectifs pédagogiques suivants :

Acquis d'Apprentissage Visés (AAV) :

A l'issue de la formation, tous les étudiants seront capables de :

- **AAV1** : Développer du code en C dans un environnement libre (sous linux en mode terminal, avec l'utilitaire make/makefile, et un compilateur croisé GCC pour risc-v).
- **AAV2** : Interfacer un périphérique sur le bus d'un softcore RISC-V (par ex. picorv32) décrit en verilog et le programmer en C.
- **AAV3** : Simuler un système modélisé en verilog en utilisant un logiciel libre (iverilog), sous linux.
- **AAV4** : Réaliser la synthèse d'un softcore RISC-V pour un FPGA, avec sa mémoire (RAM et /ou ROM) et son contenu (code binaire).

Calendrier

Jour 1 :

Lundi 16 septembre 2024 de 8h à 11h45

- En équipe : analyser les documents fournis, lister (sur un paperboard) les mots-clés connus et inconnus, les questions en suspens et les ressources pour y répondre.
- En binôme ou trinôme : démontrer la simulation verilog d'un système à base de RISC-V.
- En binôme ou trinôme : Démontrer la synthèse d'un système à base de RISC-V sur un FPGA.

Inter-séance 1-2 (livrable) :

- Individuellement : rédiger un tutoriel sur le langage verilog (notamment les différences avec le langage VHDL) et sur la simulation avec l'outil iverilog (et les différences avec l'outil ModelSim).

Jour 2 :

Lundi 23 septembre 2024 de 8h à 11h45

- En équipe: Produire son propre code C montrant l'accès à des périphériques, en modifiant leurs adresses. Adapter le code verilog aux nouvelles adresses choisies.
- En binôme ou trinôme : Tester le code C de l'équipe et montrer par simulation son bon fonctionnement.
- En équipe : Se répartir dans les 5 groupes thématiques pour l'inter-séance 2-3.

Inter-séance 2-3 :

- Produire et présenter un résumé de cours sur un des 5 thèmes : 1. Linux en mode terminal, les commandes indispensables (par ex. man, pwd, cd, mkdir, cat, more, ps, grep, / . ~ > < | etc.). 2. Utilitaire make. 3. Verilog-hdl, 4. Bus système du picorv32 et cycle bus. 5. L'accès aux périphériques mappés en mémoire en langage C. 6. Le jeu d'instruction du RISCV, les grandes familles d'instructions et modes d'adressage.

Jour 3 :

Lundi 30 septembre 2024 de 8h à 11h45

- En groupe thématique : préparer un cours (sur un paperboard) sur le thème.
- En groupe thématique : Présenter le cours préparé à l'ensemble du groupe.
- Évaluation formative (blanche) sur feuille de l'**AAV2** : « Interfacer un périphérique sur le bus d'un softcore RISC-V (par ex. picorv32) décrit en verilog et le programmer en C . »

Inter-séance 3-4 :

- Finaliser la réponse détaillée à l'évaluation formative.

Jour 4 :

Lundi 7 octobre 2024 de 8h à 11h45

- Correction de l'évaluation formative.
- **Évaluation pratique individuelle** : Produire son propre code C montrant l'accès à des périphériques, en utilisant leurs adresses. Adapter le code verilog aux adresses choisies. Tester le code C et montrer par simulation son bon fonctionnement. Réaliser la synthèse sur FPGA.
- Bilan de la formation.

Documents ressources :

Document 1 : L'interface bus du softcore RISC-V picorv32.

Document 2 : Les cycles bus du softcore RISC-V picorv32.

Document 3 : L'accès aux registres en mémoire en langage C.

Document 4 : Le fichier makefile

Document 5: Cours d'introduction au langage Verilog,

Document 6: Le système complet et son décodage d'adresse, en verilog

Document 7: Le schéma du système complet

L'ensemble de ces documents sur github :

https://github.com/JOKleinGe1/Module_Initiation_Riscv.git

Démonstration de la simulation en vidéo : <https://youtu.be/eN36onBk7ro>

Codes sources

- Pour la simulation seulement (icarus verilog) :
https://github.com/JOKleinGe1/min_sys_riscv.git
- Pour la simulation (icarus verilog) et la synthèse avec quartus (intel-fpga) :
https://github.com/JOKleinGe1/riscv_quartus.git

Paramètres de la formation (ne pas imprimer)

Jour1 : Lundi 16 septembre 2024 de 8h à 11h45

Jour2 : Lundi 23 septembre 2024 de 8h à 11h45

Jour3 : Lundi 30 septembre 2024 de 8h à 11h45

Jour4 : Lundi 7 octobre 2024 de 8h à 11h45