

## TP 3 : Simulateur de jeu d'instruction

L'objectif de ce TP est de réaliser un simulateur de jeu d'instruction pico-risc en C. Ce TP sera noté, vous enverrez vos réponses (lien git ou archive) par mail à [dylan.leothaud@irisa.fr](mailto:dylan.leothaud@irisa.fr)

Un simulateur de jeu d'instructions (Instruction Set Simulator, ou *ISS*) est un programme permettant de simuler le comportement d'un processeur. Ce programme peut être vu comme un interpréteur du jeu d'instruction. Pour simplifier ce TP, nous vous avons fourni un squelette de code à compléter. Le pseudo-code simplifié de la boucle que vous devez compléter peut être vu ainsi :

```
while (1) {  
    instruction = mem[pc];  
    opcode, rd, rs, imm = decode(instruction);  
    switch (opcode) {  
        case IMM: // execute imm  
        case ADD: // execute add  
        ...  
    }  
}
```

Pour vous permettre de tester votre simulateur de jeu d'instruction, nous vous avons également fourni une aide Javascript, *picorisc.html*, vous permettant de concevoir un fichier exécutable par votre simulateur. Pour chaque question, vous créez un ou des fichiers testant les instructions ajoutées.

Question 1. Implémenter une fonction d'extension de signe, de signature `uint16_t sext(uint8_t imm)`. Une fonction d'extension signe de 8 à 16 bits propage le bit de signe de l'immédiat sur tous les bits de l'octet de poids fort. Utilisez cette fonction pour ajouter les instructions d'opcode 0, 2, 8 et 9 (voir aide-picorisc) à votre simulateur.

Question 2. Ajouter les autres instructions arithmétiques à votre simulateurs.

Question 3. Ajouter les instructions d'accès mémoires à votre simulateur.

Question 4. Ajouter les instructions de branchement à votre simulateur.