

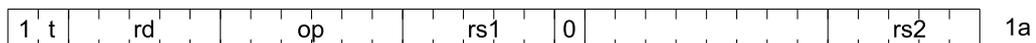
Travaux pratiques d'architecture des ordinateurs

Processeur CRAPS : le séquenceur

1. Format des instructions de CRAPS

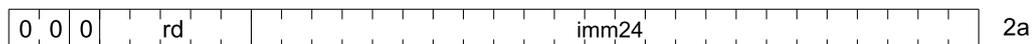
Les instructions de CRAPS ont le format binaire suivant :

Format 1: instructions arithmétiques, accès mémoire



t = 0 : instruction arithmétique, t = 1 : accès mémoire

Format 2 : sethi et branchements

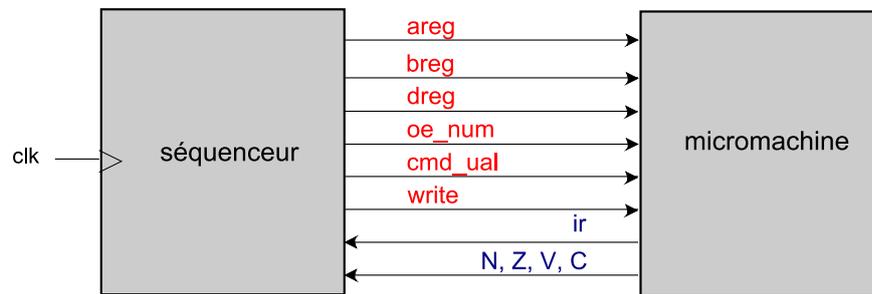


Les champs *op* et *cond* ont les significations suivantes :

| op (t=0) | Instr. | op (t=1) | Instr. | cond | branchement |
|----------|--------|----------|--------|------|-------------|
| 000000 | add | 000000 | ld | 1000 | ba |
| 010000 | addcc | 000100 | st | 0001 | be |
| 000100 | sub | | | 1001 | bne |
| 010100 | subcc | | | 0101 | bcs |
| 011010 | umulcc | | | 1101 | bcc |
| 000001 | and | | | 1110 | bpos |
| 010001 | andcc | | | 0110 | bneg |
| 000010 | or | | | 0111 | bvs |
| 010010 | orcc | | | 1111 | bvc |
| 000011 | xor | | | 1010 | bg |
| 010011 | xorcc | | | 0010 | ble |
| 001101 | slr | | | 1011 | bge |
| 001110 | sll | | | 0011 | bl |
| | | | | 1100 | bgu |
| | | | | 0100 | bleu |

2. Le séquenceur, rôle et interface

Le séquenceur envoie automatiquement les microcommandes à la micromachine, au vu de l'instruction à exécuter contenue dans %ir et de la valeur des flags N,Z,V,C



L'interface du séquenceur est donc :

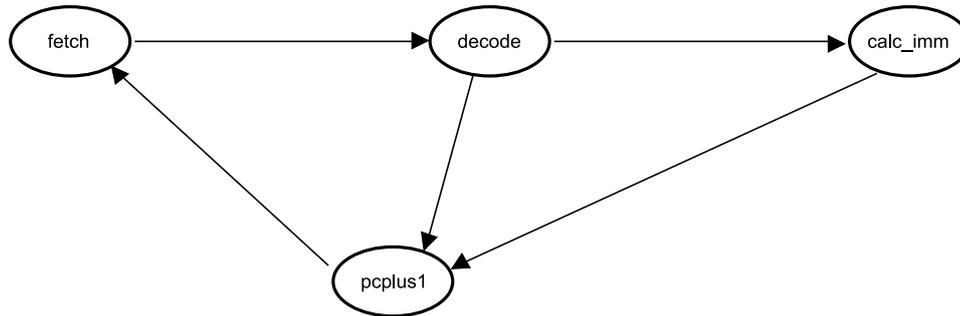
```
module sequencer(rst, clk, ir[31..0], N, Z, V, C : areg[4..0],
breg[4..0], dreg[4..0], oe_mem, cmd_ual[5..0], write)
```

L'ensemble des deux forme le processeur :

```
module craps(rst, clk, switches[15..0] : leds[15..0])
    sequencer(rst, clk, ir[31..0], N, Z, V, C : areg[4..0],
breg[4..0], dreg[4..0], oe_mem, cmd_ual[5..0], write)
    micromachine(rst, clk, areg[4..0], breg[4..0], dreg[4..0],
cmd_ual[5..0], oe_mem, write, switches[15..0] : dbus[31..0], N, Z,
V, C, ir[31..0], leds[15..0])
end module
```

1. Séquencement des instructions de calcul

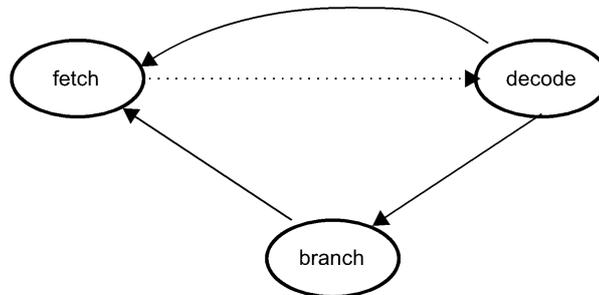
Compléter et tester le séquenceur, pour qu'il gère le sous-graphe suivant :



Tester directement le module 'craps' avec le simulateur et le vecteur de tests 'calc.tst' avec le fichier mémoire 'calc.rom'

2. Séquencement des instructions de branchement

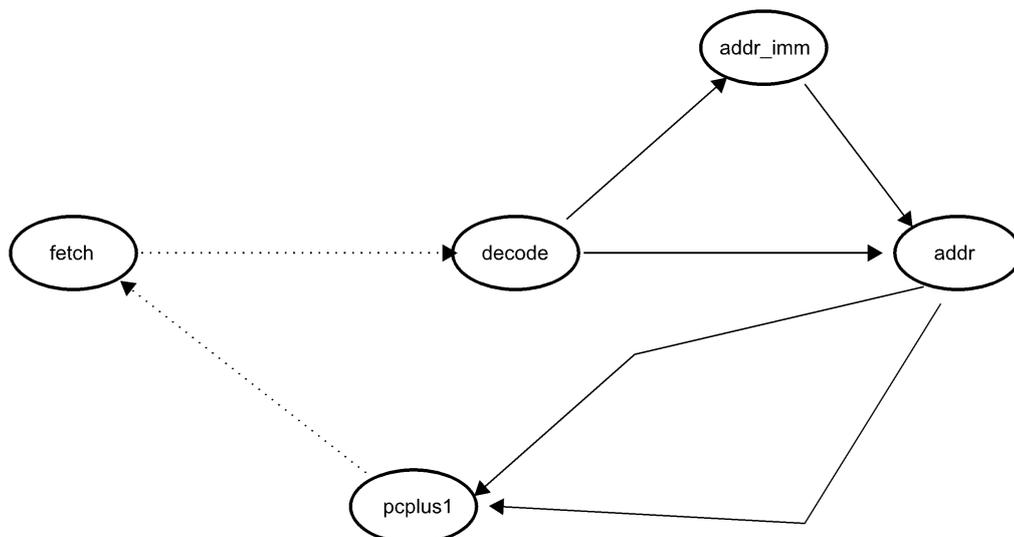
Compléter et tester le séquenceur, pour qu'il gère le sous-graphe suivant :



Tester directement le module 'craps' avec le simulateur et le vecteur de tests 'branch.tst' avec le fichier mémoire 'branch.rom'

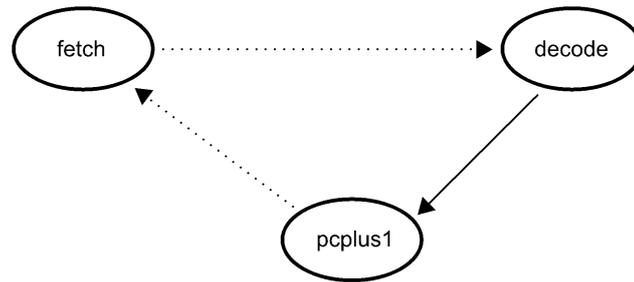
3. Séquencement des instruction de lecture et d'écriture en mémoire

Compléter et tester le séquenceur, pour qu'il gère le sous-graphe suivant :



4. Séquencement de l'instruction sethi

Compléter et tester le séquenceur, pour qu'il gère le sous-graphe suivant :



Attention au format de sethi : le champ rd n'est pas exactement à la même place que dans les instructions de calcul et d'accès mémoire.

Tester directement le module 'craps' avec le simulateur et le vecteur de tests 'squares.tst' avec le fichier mémoire 'squares.rom'