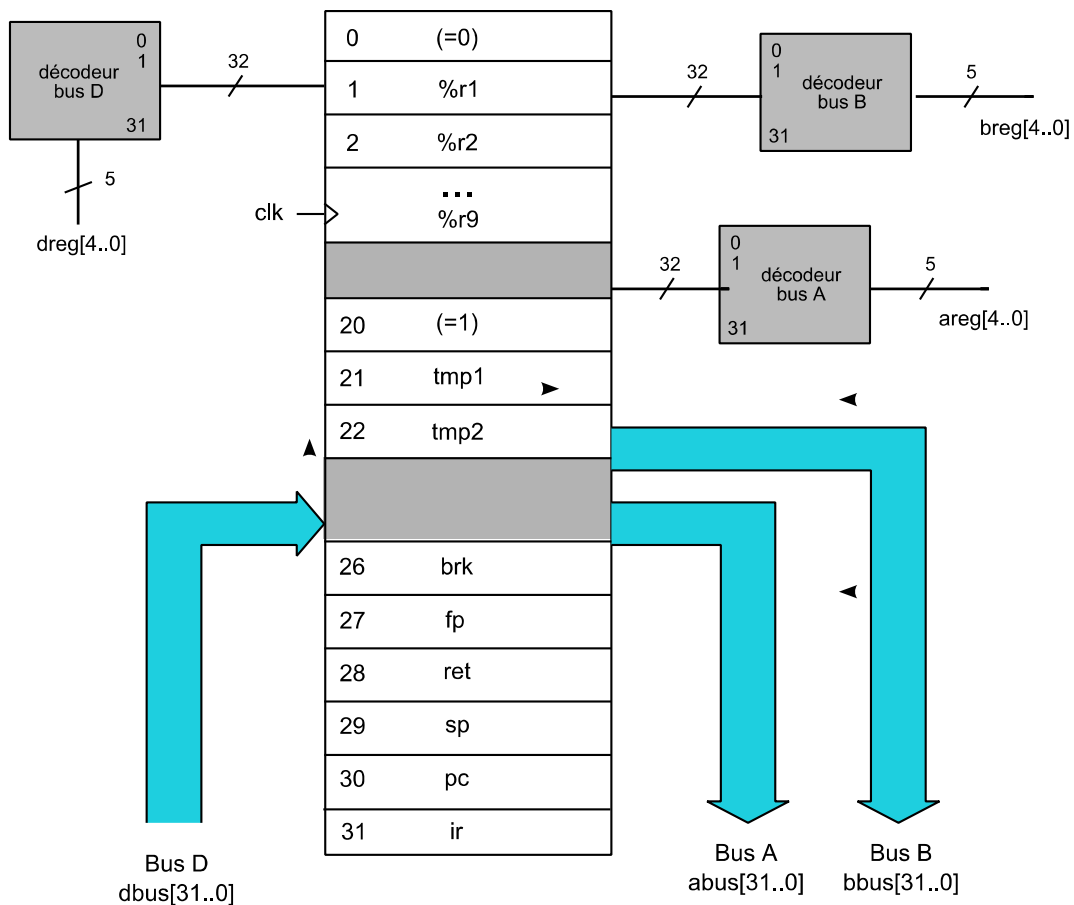


Travaux pratiques d'architecture des ordinateurs

Processeur CRAPS : le sous-système des registres

Le sous-ensemble des registres a la structure générale suivante :



1. Registre 32 bits

Créer un registre 32 bits à l'interface suivant :

```
module reg32(rst, clk, en, d[31..0] : q[31..0])
```

Au front montant de l'horloge et lorsque en vaut 1, d[31..0] est mémorisé dans q[31..0].

2. Sous-ensemble des registres de CRAPS

Concevoir le sous-ensemble des registres sous la forme d'un module SHDL d'interface :

```
module registres(rst, clk, areg[4..0], breg[4..0], dreg[4..0],
  dbus[31..0] : abus[31..0], bbus[31..0], ir[31..0], break)
```

L'écriture dans un registre est synchrone: on place la donnée à écrire sur le bus D, le numéro du registre sur dreg et l'écriture est faite lors du front montant suivant de l'horloge clk.

La lecture de la valeur d'un registre est asynchrone, et on peut lire simultanément deux registres différents sur les bus A et B. Par exemple, on place le numéro du registre à lire sur areg, et la valeur apparaît immédiatement sur le bus A, sans attendre le front d'horloge. Idem avec breg et le bus B.

On intégrera :

- un nombre limité de registres banalisés, par exemple %r1.. %r9
- les registres utilisés par le processeur, de numéros 21 à 22 et 26 à 31
- les registres constants %r0 (=0) et %r20 (=1)

La sortie break vaut 1 lorsque la valeur du registre brk égale la valeur du registre pc; on utilisera le module de librairie eq32.

Tester le module avec les entrées-sorties à distance, en vérifiant qu'on peut bien écrire et lire dans chaque registre, et que les registres constants donnent les bonnes valeurs.

Note: on utilisera des bus 3-états pour abus et bbus afin d'avoir l'écriture la plus modulaire possible.