

Contrôle d'un processeur pipeline

Stéphane Rubini

On considère le chemin de données de processeur pipeline représenté figure 2. Il s'agit de la version "pipelinée" 5 étages du chemin de données séquentiel étudié lors du TP précédant; les signaux de contrôle de celui-ci sont les mêmes.

L'unité de contrôle produit les signaux de contrôle pour le chemin de données et quelques signaux supplémentaires décrits ci-dessous.

— **Bus_Val** : bus de données et adresses valides

— **RW** : transfert en lecture (1) ou écriture (0)

L'unité de contrôle reçoit aussi des signaux en entrée :

— **Inst/32** : bus d'instructions (instructions lues en mémoire)

— **Clk** : horloge de l'ensemble du processeur

— **Z,N** : indicateurs d'état en provenance de l'UAL

Le codage des instructions est simplifié au maximum car toutes sont sur le même format 32 bits :

position	31	28	27	24	23	20	19	16	15	0
champs	codeop		reg src1		reg src2		reg dest		immediat	

Travail demandé Dans ce TP, il s'agit de développer l'unité de contrôle d'un processeur RISC. Le chemin de données, le modèle structurel du processeur (qui utilisera votre unité de contrôle) et son banc de test vous sont fournis (cf Fig. 1).

1. Modéliser en VHDL l'unité de contrôle complète (pour les codes opération, voir le tableau 1).
2. Simuler le processeur à l'aide du banc de test fourni.
3. Intégrer dans le processeur un mécanisme de traitement des aléas de données de niveau 1 (court-circuit).

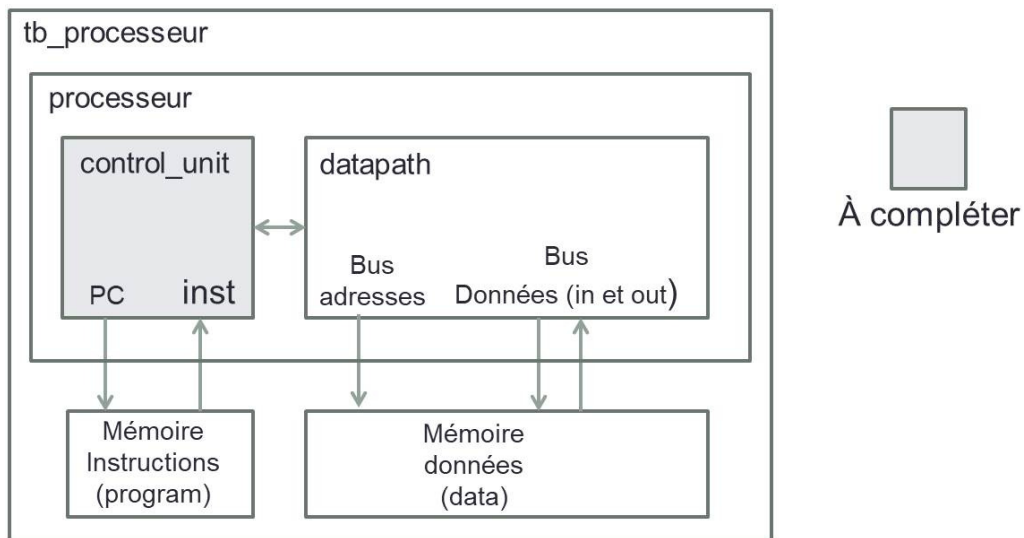


FIGURE 1 – Structure du modèle VHDL.

codeop	mnémonique assembleur
0000	NOP
0001	ADDI
0010	SUB
0011	SW
0100	LW
0101	JSR
0110	RTS
0111	BEQ
1000	BRA

TABLE 1 – Code opération (codeop) des instructions

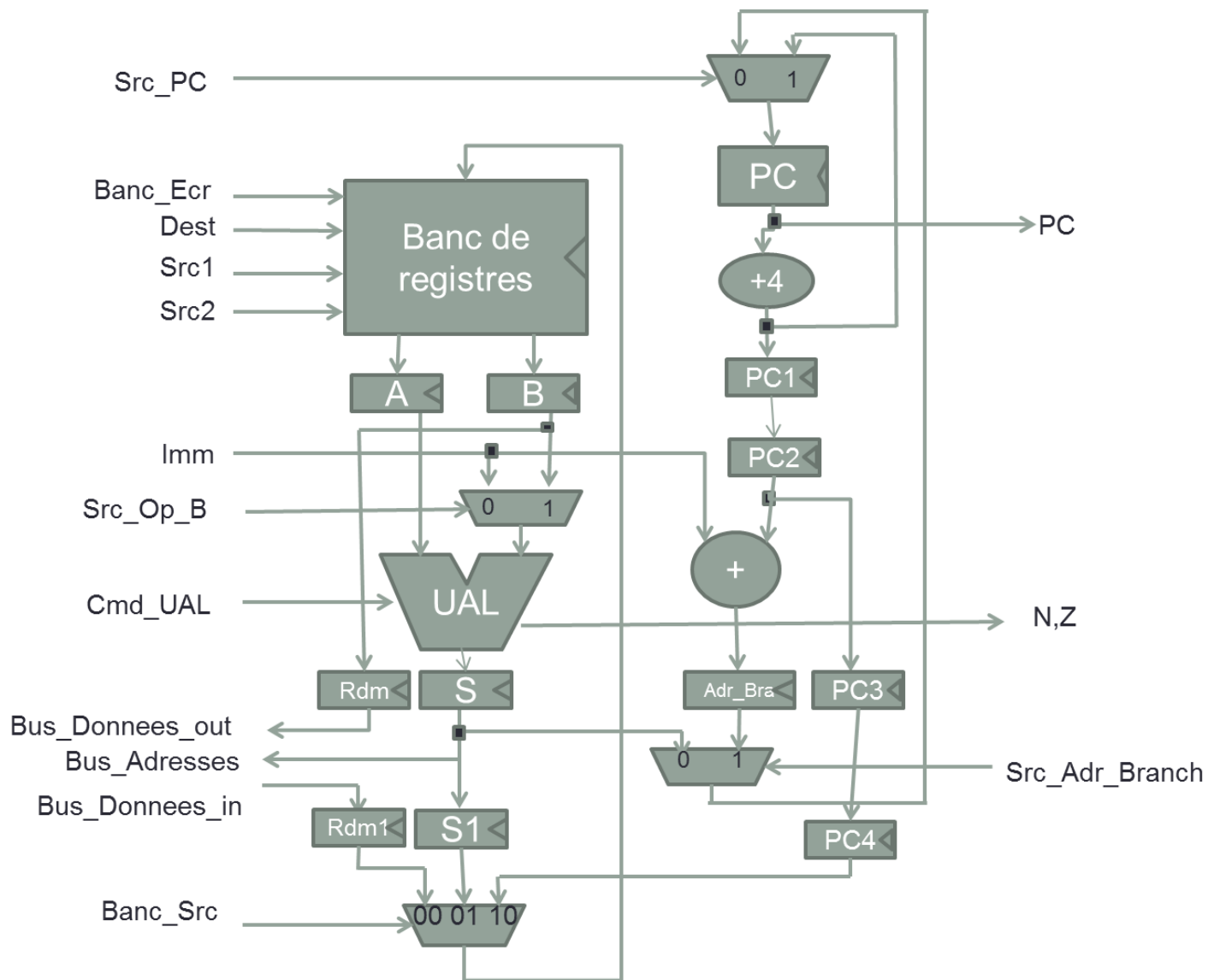


FIGURE 2 – Chemin de données pipeliné du processeur.