

Les composantes du processeur avec pipeline sans gestion des aléas

Cette section du document doit être lue en visualisant le circuit le plus complet du processeur avec pipeline sans gestion des aléas présenté en classe afin de bien comprendre les explications comprises dans ce document.

Si une composante du processeur n'est pas décrite dans cette section, veuillez vous référer au document sur le processeur à 1 cycle. Cela signifie que la composante fonctionne de la même façon. En fait, toutes les composantes du processeur (à l'exception des registres inter-étapes) sont identiques à celles du processeur à 1 cycle.

Les registres

IF/ID

Registre pour la communication entre les étapes *Instruction fetch* et *Instruction decode*. Contient :

- l'adresse de la prochaine instruction
- l'instruction chargée en mémoire

ID/EX

Registre pour la communication entre les étapes *Instruction decode* et *Execute*. Contient :

- les bits de contrôle pour les étapes *Execute*, *Memory* et *Write-Back*
- l'adresse de la prochaine instruction
- la valeur du premier registre source
- la valeur du deuxième registre source
- la valeur immédiate de l'instruction étendue sur 32 bits
- le champ *rd* de l'instruction de type R
- le champ *rt* de l'instruction de type I

EX/MEM

Registre pour la communication entre les étapes *Execute* et *Memory*. Contient :

- les bits de contrôle pour les étapes *Memory* et *Write-Back*
- le *Branch Target*
- le résultat de l'UAL
- le bit zéro
- la valeur du deuxième registre source
- le numéro du registre de destination

MEM/WB

Registre pour la communication entre les étapes *Memory* et *Write-Back*. Contient :

- les bits de contrôle pour l'étape *Write-Back*
- la valeur lue en mémoire
- le résultat de l'UAL
- le numéro du registre de destination

Les composantes du processeur avec pipeline avec gestion des aléas

Cette section du document doit être lue en visualisant le circuit le plus complet du processeur avec pipeline avec gestion des aléas présenté en classe afin de bien comprendre les explications comprises dans ce document.

Les nouvelles composantes

=

Une nouvelle composante a été ajoutée à la sortie du banc de registres afin d'effectuer immédiatement la comparaison entre les deux valeurs provenant du banc de registres. Le résultat de cette comparaison servira éventuellement à indiquer si un branchement doit être effectué si nous exécutons une instruction *beq*.

Hazard detection unit

Cette unité sert à ajouter des bulles dans le pipeline. Les entrées de cette unité servent à indiquer s'il est nécessaire d'ajouter une bulle dans le pipeline. Un signal part de l'unité pour aller vers les registres PC et IF/ID. Ce signal indique si l'écriture dans ces registres est permise. Lorsque l'unité voudra insérer une bulle dans le pipeline, l'écriture ne sera pas permise dans les registres PC et IF/ID durant un cycle. Un autre signal se dirige vers un multiplexeur qui servira à transformer les bits de contrôle en 0. Ceci aura pour effet de transformer n'importe quelle instruction en *nop*. Également, lorsqu'une prédiction de branchement est erronée, il est nécessaire de vider le registre IF/ID car elle contiendra une mauvaise instruction. Ceci est fait à l'aide du signal IF.Flush.

Forwarding unit

Cette unité détecte les dépendances entre les instructions qui se suivent dans le pipeline et permet d'acheminer des nouvelles valeurs de registre vers les instructions qui en dépendent avant même que la valeur ne soit stockée réellement dans le banc de registres.