

1. Contrôle préalable

Cette partie est conçue comme une vérification pour vous permettre de déterminer si vous comprenez les concepts abordés en cours ou non. Veuillez répondre par « Vrai » ou « Faux » aux questions suivantes et inclure une explication :

- 1.1. Le chemin de données à cycle unique utilise toutes les unités matérielles pour chaque instruction.

Faux ! Toutes les unités sont actives dans chaque cycle, mais certaines sorties peuvent être ignorées par des signaux de contrôle des multiplexeurs.

- 1.2. Il est possible d'exécuter les étapes du chemin de données à cycle unique en parallèle pour accélérer l'exécution d'une seule instruction.

Faux ! Chaque étape dépend de la valeur produite par l'étape précédente (par exemple, l'étape de décodage d'instructions dépend de l'instruction récupérée depuis la mémoire).

- 1.3. Les circuits combinatoires ne sont utilisés que dans l'étape de décodage des instructions.

Faux ! D'autres étapes exécutent également des circuits combinatoires (multiplexeurs pour l'extraction d'instructions ; opérations UAL pendant l'étape d'exécution, ...).

2. Processeur à cycle unique

- 2.1. Dans ce TD, nous travaillerons avec le chemin de données du processeur à cycle unique disponible sur la dernière page.

- a) Sur le chemin de données, remplissez chaque case arrondie avec le nom du composant du chemin de données et chaque case carrée avec le nom du signal de commande.
- b) Expliquez ce qui se passe à chaque étape du chemin de données.

IF Instruction Fetch (Récupération d'instruction)

Envoie l'adresse à la mémoire d'instructions (IMEM) et lit IMEM à cette adresse

ID Instruction Decode (Décodage d'instruction)

Génère les signaux de contrôle à partir de bits de l'instruction. Génère la constante « Immédiat », et lit les contenus de registres à partir du « banc de registres ».

EX Execute (Exécution de l'instruction)

Effectue les opérations UAL et des comparaisons pour les branchements.

MEM Memory (Accès Mémoire)

(Lit à partir / Ecrit dans) la mémoire de données (DMEM).

WB Writeback (Transcription)

Transcrit dans un registre le résultat de l'UAL ou la donnée retournée par la mémoire des données (DMEM).

2.2. Ajoutez les fils de données et les signaux de contrôle pour le circuit calculant le prochain PC.

2.3. Implémentez la logique du circuit qui calcule le prochain PC.

2.4. Remplissez le tableau suivant avec les signaux de contrôle pour les instructions données en se basant sur le chemin de données que vous avez complété. Dans le cas où la nature du signal de contrôle n'a pas d'importance pour l'instruction examinée, mettez un **X** dans la case correspondante.

Instrs.	Signaux de contrôle								
	Jump	RegDst	ImmSel	RegWEn	BSEL	ALUSel	MemRW	WBSEL	Branch
add	0	1	X	1	0	add	0	0	0
ori	0	0	0	1	1	or	0	0	0
lw	0	0	1	1	1	add	0	1	0
sw	0	X	1	0	1	add	1	X	0
beq	0	X	1	0	0	sub	0	X	1
j	1	X	X	0	X	XXXX	0	X	X

2.5. Un circuit d'état est un composant connecté à l'horloge (désigné par un triangle en bas du circuit). Afin que le composant fonctionne correctement, le signal en entrée doit être stable avant chaque front montant de l'horloge.

Le chemin critique décrit la période nécessaire à l'exécution de tous les éléments d'état présents dans un circuit. Le circuit ne peut pas être cadencé plus rapidement que cette période parce que la stabilité du signal n'est plus garantie dans ce cas.

Pour cet exercice, supposons que le délai pour chaque étape du chemin de données est le suivant (les durées sont données en picosecondes – $1 \text{ ps} = 10^{-12} \text{ seconde}$) :

IF : 200 ps ; **ID** : 100 ps ; **EX** : 200 ps ; **MEM** : 200 ps ; **WB** : 100 ps

a) Pour chaque instruction citée dans le tableau ci-dessous, indiquez par un X les étapes du chemin de données utilisées et calculez le temps minimum nécessaire pour son exécution.

	IF	ID	EX	MEM	WB	Durée totale
add	X	X	X		X	600 ps
ori	X	X	X		X	600 ps
lw	X	X	X	X	X	800 ps
sw	X	X	X	X		700 ps
beq	X	X	X			500 ps
j	X	X				300 ps

- b) Quelle(s) instruction(s) exercent le chemin critique ?

L'instruction `lw` ! Elle utilise les cinq étapes du chemin de données. (voir la ligne en rouge dans le diagramme).

- c) En se basant sur votre réponse précédente, à quelle fréquence peut-on donc cadencer ce processeur à cycle unique ?

$$\frac{1}{800 \text{ ps}} = \frac{1}{800 \times 10^{-12} \text{ sec}} = 1\,250\,000\,000 \text{ Hz} = 1,25 \text{ GHz}.$$

- d) Pourquoi le chemin de données à cycle unique est inefficace ?

Dans un chemin de données à cycle unique, la plupart des étapes du chemin sont le plus souvent dans un état oisif (c.-à-d. se tournent les pouces à ne rien faire). De plus, même si peu d'instructions requiert l'exécution de toutes les étapes (c.-à-d. le chemin critique), les instructions qui peuvent être traitées avec moins d'étapes dans le chemin de données sont cadencées à la vitesse de l'instruction la plus lente.

- e) Comment peut-on améliorer les performances ? Quel est le but du pipelining ?

Les performances peuvent être améliorées avec le pipelining (en plaçant des registres entre les étapes). Nous traiterons cela en détails dans le prochain TD.

