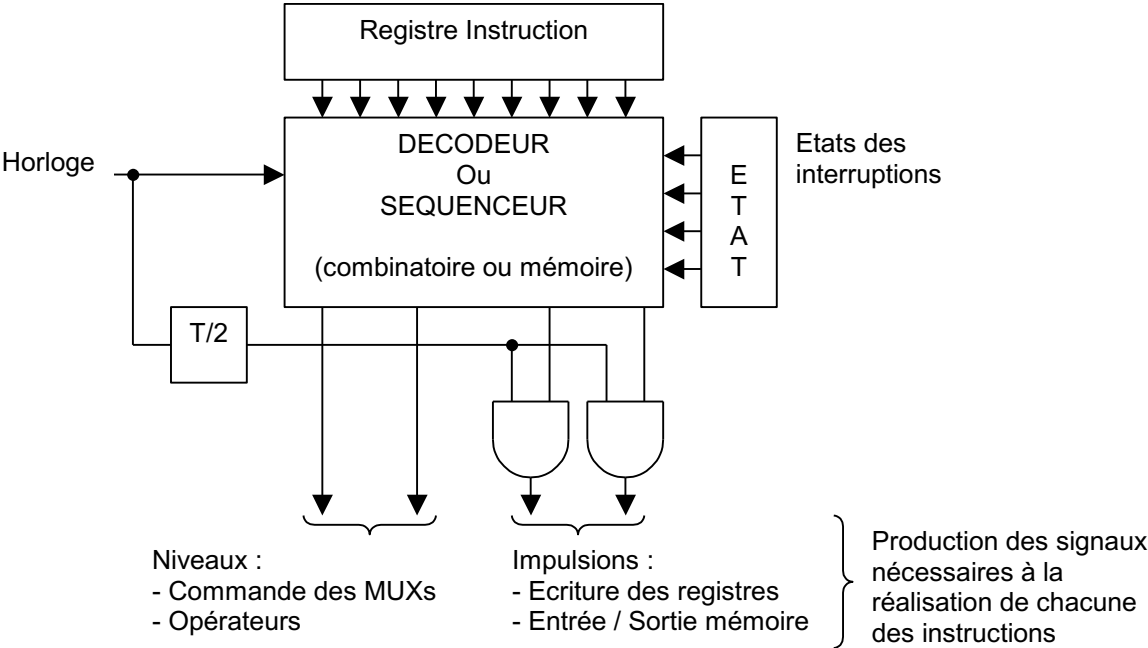
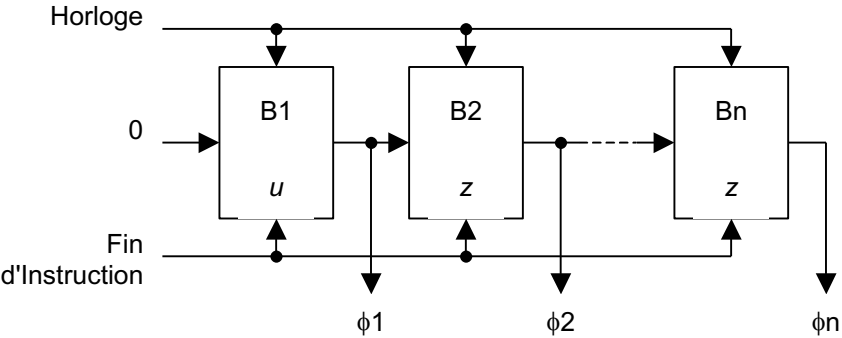


TD 4 *Architecture*

DECODAGE DES INSTRUCTIONS



Le marquage du temps indique comment déterminer à quelle phase d'une séquence d'instruction on se trouve. C'est à dire comment construire les signaux ϕ_1, \dots, ϕ_n qui déclenchent les séquences d'instruction.



u est une entrée qui force la bascule à 1 indépendamment de l'horloge.
z est une entrée qui force la bascule à 0 indépendamment de l'horloge.

Exercice 1 : Décodeur ou Séquenceur Câblé et Microprogrammé

Supposons que le format du registre d'instruction soit :

Code Opération				Mode d'Adressage	
A	B	C	D	E	F
Référence Adressage					

Codage des opérations :

A	B	C	D	Opération
0	0	0	0	NOP
0	0	0	1	LOAD A
0	0	1	0	ADD B
0	0	1	1	JUMP
0	1	0	0	JUMP C
0	1	0	1	RETURN
0	1	1	0	CALL

Codage des modes d'adressage :

E	F	G	Mode d'Adressage
0	0	0	Immédiat
0	0	1	Immédiat étendu
0	1	0	Direct
0	1	1	Direct étendu
1	0	0	Indirect
1	0	1	Indirect étendu
1	1	0	Relatif
1	1	1	Relatif étendu

Remplir le tableau en considérant uniquement des adressages étendus sur l'architecture S1.

Séquenceur câblé :

Le séquenceur est un circuit combinatoire multi-sortie.

Donner les équations booléennes de chaque signal du tableau.

COB = ?

Il faut faire la même démarche pour les autres signaux.

Instruction Code	No Opération	LOAD A, IMM Et.	LOAD A, DIR	LOAD A, REL	ADD B, DIR Et.
Phase 1					
Phase 2					
Phase 3					
Phase 4					
Phase 5					
Phase 6					
Phase 7					
Phase 8					
Phase 9					
Phase 10					
Phase 11					
Phase 12					

Instruction Code	JUMP DIR	JUMP C, REL		CALL DIR	RETURN DIR
		Condition Fausse	Condition Vraie		
Phase 1					
Phase 2					
Phase 3					
Phase 4					
Phase 5					
Phase 6					
Phase 7					
Phase 8					
Phase 9					
Phase 10					
Phase 11					
Phase 12					

Séquenceur microprogrammé :

Le séquenceur comporte une mémoire dite de "microprogramme" qui est lue à chaque période de l'horloge.

Une mémoire contient, pour chaque instruction et pour chaque phase de l'instruction, un mot dans lequel est codé les signaux.

Un nouveau mot est lu à chaque période d'horloge et sa sortie sur les lignes produit le signal.

- Le codage du type d'opération et du mode d'adressage des instructions nous donne les différentes adresses du microprogramme.

Exemple : ADD B, Direct étendue
 0010 011 -> 19

- Le cycle FETCH est commun à toutes les instructions. Il faut le placer à une certaine adresse du microprogramme et aller le chercher avant les instructions.
- ϕ_1 , ϕ_2 , et ϕ_3 constituent le cycle FETCH. Donc c'est ϕ_4 qui commencera l'instruction.
- Lorsqu'une instruction est terminée, ne pas oublier de valider FIN. Le signal FIN permet de gérer les cycles de phases et donc de remettre ϕ_1 à 1.
- Sélection des adresses de micro-instructions
 - Durant la première période d'horloge l'adresse de début du FETCH est chargée dans RAMM.
 - Ensuite ϕ_1 prend la valeur 0 et la garde jusqu'à début de l'instruction suivante. Alors, la sélection des adresses dépend de SelMS.

SelMS

- 1 Adresse = Adresse + 1
- 2 Adresse dépend de la condition
 - condition = 0 -> Adresse = Adresse + 1
 - condition = 1 -> Adresse = Adresse contenue dans la micro-instruction précédente
- 3 Adresse = valeur du Code Opération de l'instruction
- 4 Adresse = Adresse contenue dans la micro-instruction précédente

A l'aide de ces indications, remplir le tableau 2 du séquenceur microprogrammé distribué.

