

NOM, Prénom :

# DS Architecture des Circuits

21 décembre 2023

Durée 1h30.

Répondez sur le sujet.

REMP LISSEZ VOTRE NOM TOUT DE SUITE

Vingt questions, un point par question quelle qu'en soit la difficulté ; pas de points négatifs.

Tout dispositif électronique interdit (calculatrice, tablette, ordinateur, montre connectée, écouteurs...)

Un seul document autorisé : une feuille A4 recto-verso de notes personnelles.

Crayon à papier accepté, de préférence aux ratures et surcharges. Pour les schémas l'utilisation d'une règle n'est pas indispensable mais elle est tout de même recommandée !

**Q1.** Dessinez ici le schéma général d'implémentation d'une machine de Moore synchrone (vous veillerez à bien faire figurer tous les signaux et à nommer tous les éléments).

**Q2.** Dessinez ici le schéma général d'une machine de von Neumann (vous veillerez à bien faire figurer tous les signaux et à nommer tous les éléments).

**Q3.** Classez les six nombres hexadécimaux suivants par ordre croissant :

0x7069, 0xC709, 0xEDDD, 0x8000, 0x1000, 0x8858

(1) en supposant qu'il s'agit d'entiers naturels

--	--	--	--	--	--

(2) en supposant qu'il s'agit d'entiers relatifs codés en complément à deux

--	--	--	--	--	--

**Q4.** Sachant que les deux nombres suivants sont codés en complément à deux :

$x_1 = \text{FFFF.FFB5}$

$x_2 = \text{0000.086D}$

Quelles sont leurs valeurs respectives (en décimal) ?

Réponse :  $x_1 =$

Réponse :  $x_2 =$

**Q5.** Sachant que les nombres suivants sont codés en complément à deux sur 16 bits, calculez le résultat des quatre opérations suivantes ainsi que les Flags Z (Zero), N (Negative), C (Carry) et V (oVerflow) correspondants :

(1)  $0xFFFF + 0x0001 = 0x$                       Z =      N =      C =      V =

(2)  $0xFFFF + 0xFFFF = 0x$                       Z =      N =      C =      V =

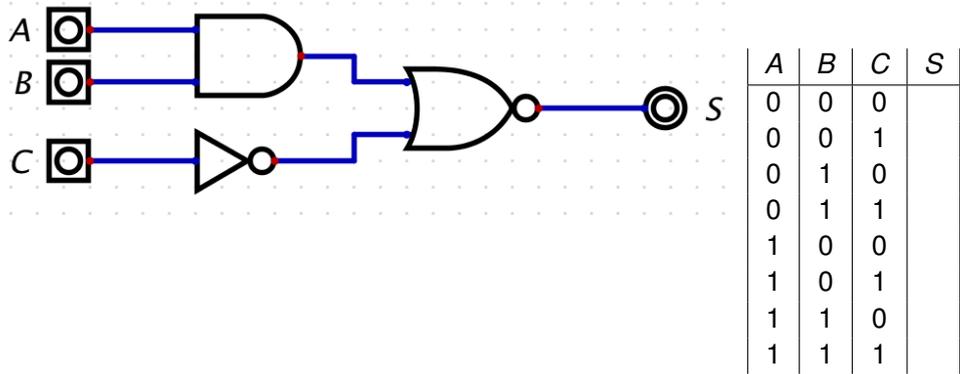
(3)  $0x8000 + 0x0001 = 0x$                       Z =      N =      C =      V =

(4)  $0x8000 + 0xFFFF = 0x$                       Z =      N =      C =      V =

**Q6.** Dessinez à droite le circuit correspondant à la table de vérité ci-dessous à gauche. Vous pouvez utiliser des portes à une, deux, trois ou quatre entrées et vous tacherez de minimiser le nombre de portes.

A	B	C	D	S
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	1

**Q7.** Donnez la table de vérité correspondant à ce circuit.



**Q8.** soit l'équation booléenne suivante :

$$x = abc\bar{d} + bca + a\bar{c}b + \bar{b}c\bar{d}b + a\bar{a} + \bar{a}bcd$$

Que vaut x dans les cas suivants :

- $a = 1, b = 0, c = 0, d = 0$  :  $x =$
- $a = 0, b = 1, c = 0, d = 0$  :  $x =$
- $a = 0, b = 0, c = 1, d = 0$  :  $x =$
- $a = 0, b = 0, c = 0, d = 1$  :  $x =$

**Q9.** On conçoit une machine de von Neumann disposant de 64 instructions. Ces 64 instructions se composent de :

- 18 pseudo-instructions (à l'instar du "NOP" vu en TP),
- 20 instructions utilisant 4 modes d'adressage,
- 4 instructions utilisant 3 modes d'adressage,

Les 22 instructions restantes n'utilisent qu'un seul mode d'adressage.

Quelle taille minimale (en nombre de bits) devons-nous réserver pour coder l'OPCODE de cette machine ?

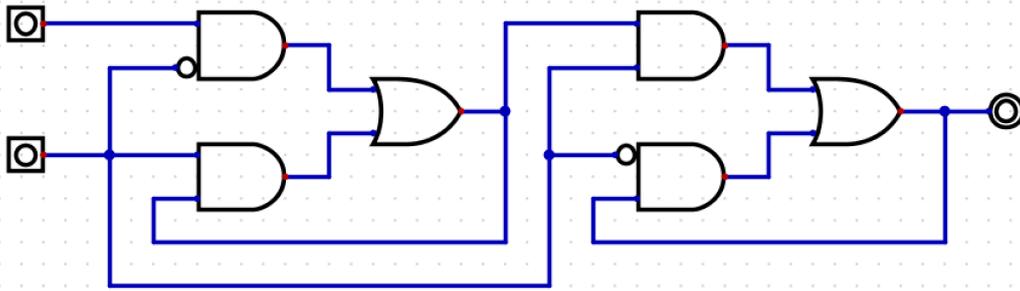
- Réponse :

Expliquez votre réponse en quelques mots :

**Q10.** Soit une mémoire de 1 048 576 mots de 8 bits.

- Donnez la capacité de cette mémoire (en octets) :
- Quelle est (en bits) la largeur de ses entrées/sorties.
  - $WE$  :
  - $A$  :
  - $D_{in}$  :
  - $D_{out}$  :
- On veut écrire la valeur 0x64 dans la 64ième "case" de cette mémoire. Quelle(s) valeur(s) hexadécimale(s) faut-il envoyer sur chacune de ses entrées/sorties ?
  - $WE$  :
  - $A$  :
  - $D_{in}$  :
  - $D_{out}$  :

**Q11.** Quelle est la fonction de ce circuit ?



— Réponse :

Compléter le schéma du circuit en indiquant, sur le schéma, un nom **cohérent** pour chacune des entrées et des sorties.

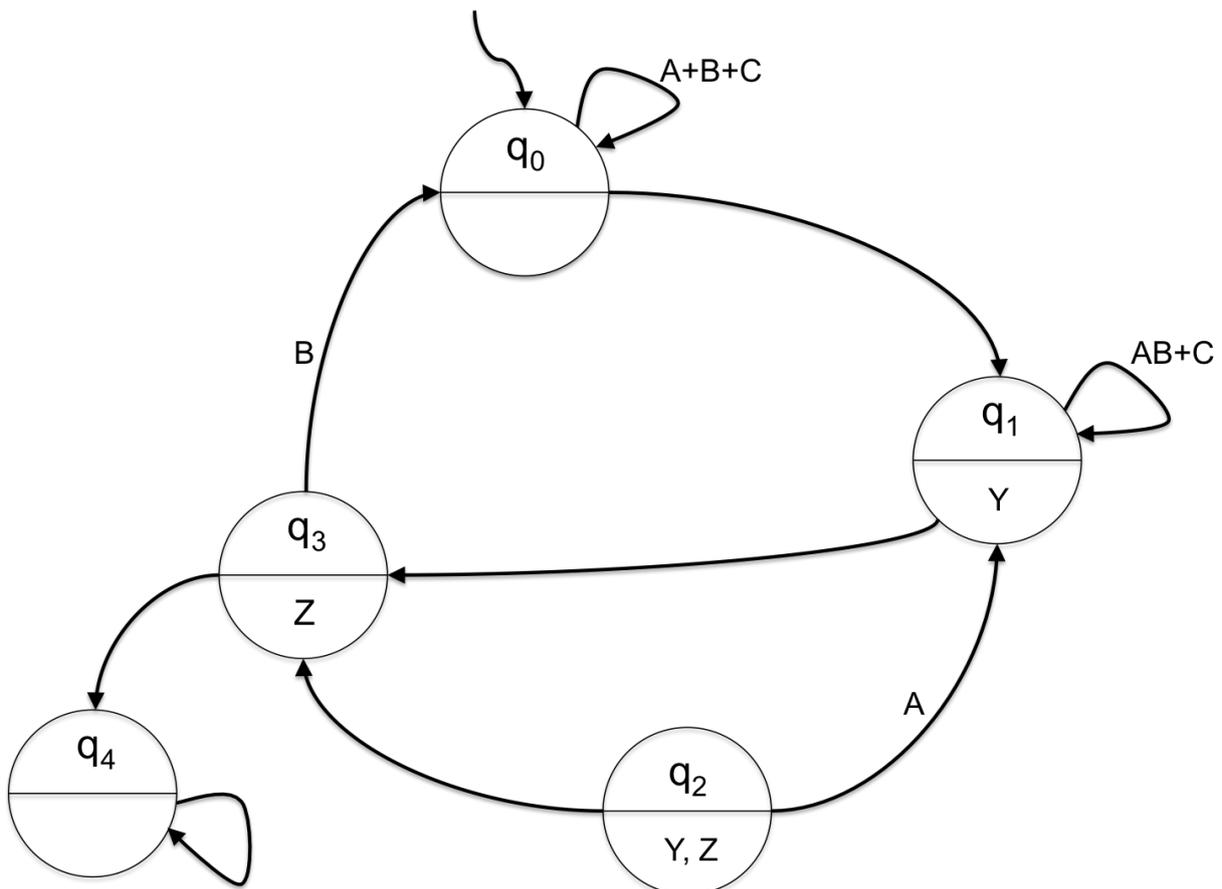
**Q12.** On considère une machine de von Neumann classique. Cette machine est en train d'exécuter un code assembleur.

- Au temps  $t = 42$ , le registre PC de la machine contient la valeur 0x0A12.
- Au top d'horloge suivant, le registre PC contient la valeur 0x0923.

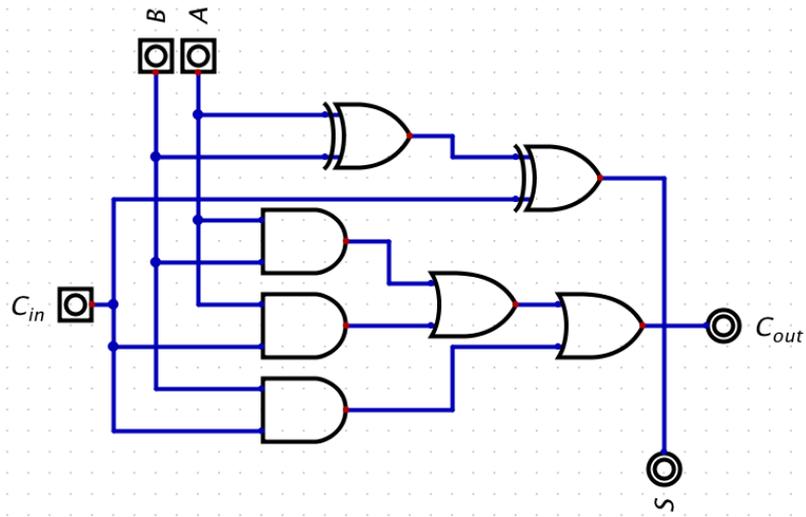
Quel est le type d'instruction en cours d'exécution au temps  $t = 42$  dans cette machine ?

— Réponse :

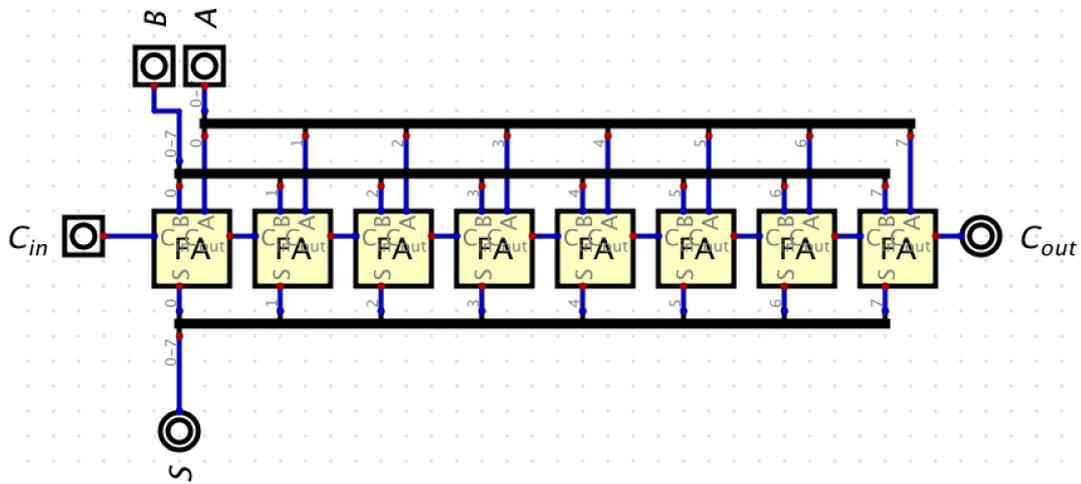
**Q13.** Dans l'automate de Moore suivant, des conditions ont été oubliées sur plusieurs transitions. Complétez-le de façon à ce qu'il soit synchrone, réactif et déterministe (sans modifier les conditions déjà mentionnées).



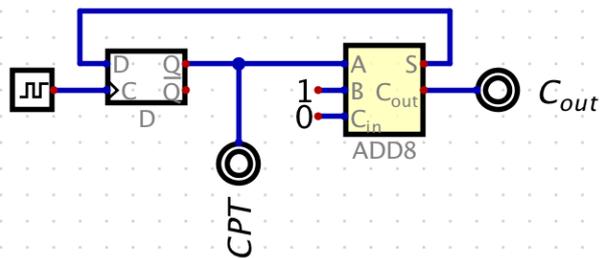
**Q14.** On considère le circuit suivant



Ce circuit (nommé "FA") est réutilisé pour implémenter le circuit "ADD8" suivant :



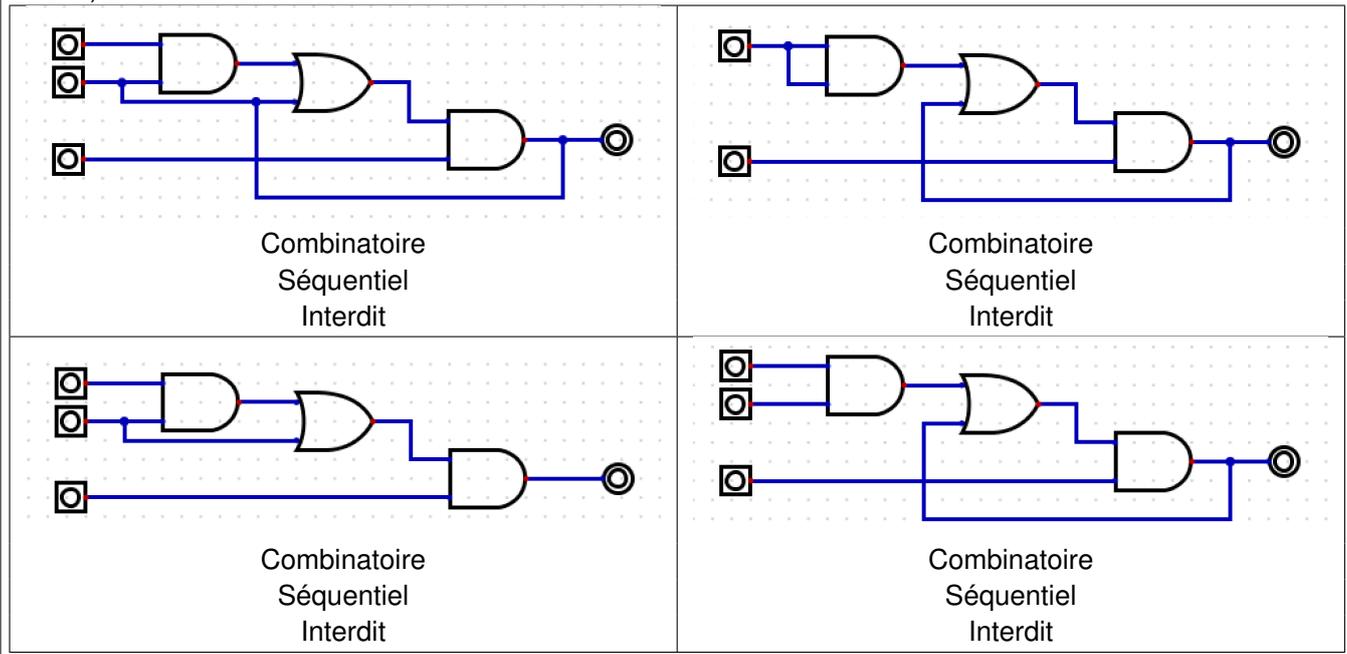
Qui est lui-même utilisé dans le compteur suivant :



On suppose que le temps de commutation des portes est  $\tau = 10^{-6}$  secondes (quel que soit le type de porte), quelle est (en secondes) la période minimale de l'horloge du compteur ?

— Réponse :

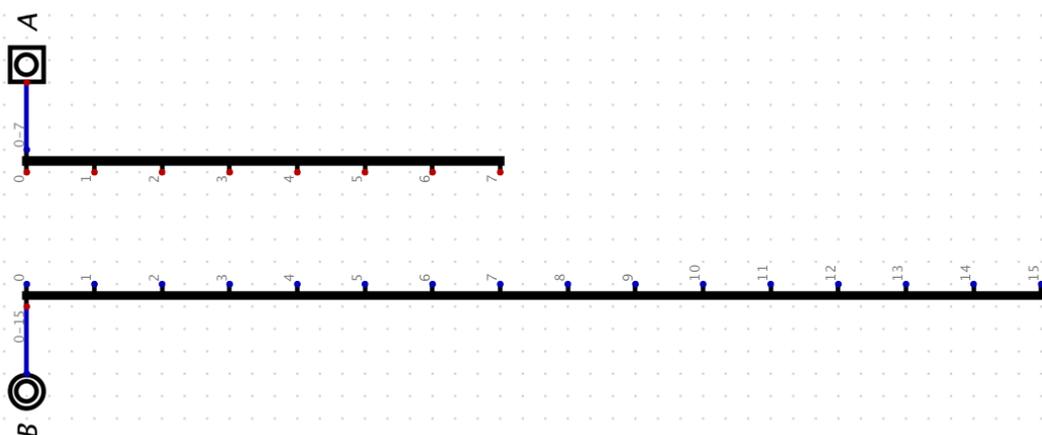
**Q15.** Parmi les quatre circuits ci-dessous, lesquels sont combinatoires, lesquels sont séquentiels, lesquels sont interdits ? (entourez les bonnes réponses ; l'absence de réponse est considérée comme une réponse fausse)



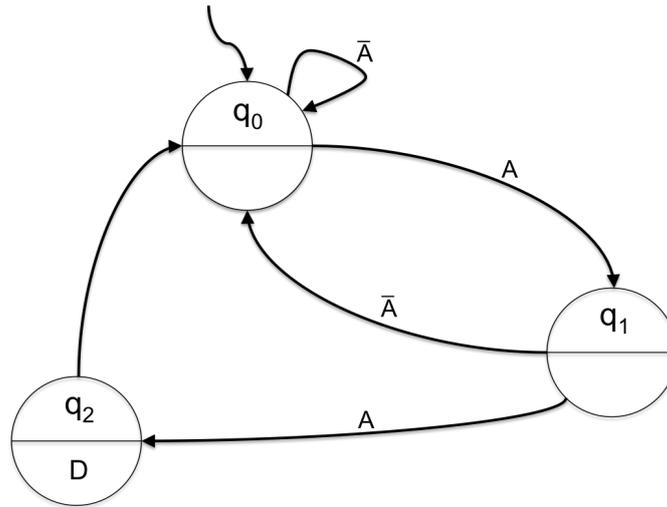
**Q16.** Simplifier l'équation booléenne suivante en détaillant les principales étapes de votre raisonnement.

$$s = \overline{\overline{(x + y)} \cdot \overline{(x + z)} \cdot \overline{(y + z)}}$$

**Q17.** Sachant que  $A$  est un entier codé en complément à deux sur 8 bits et que  $B$  est un entier codé en complément à deux sur 16 bits, complétez le circuit de façon à ce que  $B$  soit égal à  $8 \times A$ .



**Q18.** On souhaite créer un circuit numérique implémentant l'automate de Moore synchrone suivant (en utilisant un codage logarithmique des états) :



— Donner les tables de vérité correspondant à sa fonction de transition et à sa fonction de sortie

— Donner les équations booléennes de sa fonction de transition et de sa fonction de sortie

— Dessinez le circuit numérique implémentant cette FSM

**Q19.** Sachant que A et Data\_IN sont deux entiers naturels codés sur 32 bits, on souhaite créer un circuit numérique calculant la suite de Syracuse selon l'algorithme suivant :

```
A = Data_IN
Afficher A
TANT_QUE (A < 0x55555555) :
  SI A pair:
    A = A / 2
  SINON:
    A = (A * 3) + 1
  FIN_SI
Afficher A
FIN_TANT_QUE
```

Quelle est, selon vous, la signification du test (A < 0x55555555) ?

— Réponse

Dessinez ci-dessous une proposition pour le datapath de votre circuit. Vous pouvez utiliser des portes logiques, des registres, des multiplexeurs, des démultiplexeurs, des additionneurs, des soustracteurs, des décalages à gauche (SHL) et des décalages à droite (SHR) :

Dessinez ci-dessous une proposition pour l'automate de votre circuit (vous veillerez évidemment à la cohérence entre votre automate et votre datapath). Le dessin de l'automate suffit ; il n'est pas demandé de proposer une implémentation.

**Q20.** Soit une EEPROM dont le contenu est entièrement présenté dans le tableau suivant.

Addr...	0	1	2	3	4	5	6	7	8	9	0xA	0xB	0xC	0xD	0xE	0xF
0	0x7F	0x81	8	0x85	8	0x87	8	0x88	0x7F	0x89	0	0x90	0x7F	0x91	0x49	0x92
0x10	0x49	0x93	0x41	0x94	0	0x95	0x7F	0x96	0x97	0x98	1	0x99	0xA0	0xA1	0	0xA2
0x20	0x7F	0xA3	1	0xA4	1	0xA5	1	0xA6	0	0xA7	0x3E	0xA8	0x41	0xA8	0x41	0xAA
0x30	0x41	0xAB	0x3E	0xAC	0	0xAD	0	0xAE	0	0xAF	0	0xB0	0	0xB1	0	0xB2
0x40	0x78	0xB3	0xF	0xB4	1	0xB5	2	0xB6	0xC	0xB7	2	0xB8	1	0xC0	0xF	0xC1
0x50	0x78	0xC2	0	0xC3	0x3E	0xC4	0x41	0xC5	0x41	0xC6	0x41	0xC7	0x3E	0xC8	0	0xC9
0x60	0x7F	0xCA	0x48	0xCB	0x4C	0xCC	0x4A	0xCD	0x31	0xCE	0	0xCF	0x7F	0xFF	1	0xD1
0x70	1	0xD2	1	0xD3	0	0xD4	0x41	0xD5	0x7F	0x81	0x41	0x81	0x41	0x81	0x63	0x81
0x80	0x3E	0x81	0	0x81												
0x90	0	0x81														
0xA0	0x81	0x9F	0x9F	0x9F												
0xB0	0x9F															
0xC0	0x9F															
0xD0	0x9F															
0xE0	0x9F	0x7F	0x81	8	0x85	8	0x87									
0xF0	8	0x88	0x7F	0x89	0	0x90	0x7F	0x91	0x49	0x92	0x49	0x93	0x41	0x94	0	0x95

Quelle est la capacité de cette mémoire (en octets) ?

— Réponse :

Quelle est la largeur de son bus d'adresse (en bits) ?

— Réponse :

On lit cette EEPROM à l'adresse 0x73 ; Quelle valeur l'EEPROM retourne-t-elle ?

— Réponse :

