

Systèmes numériques DE

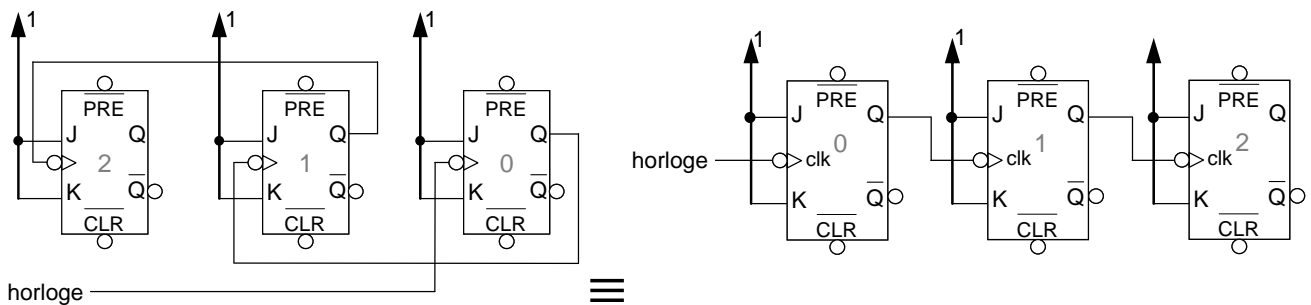
Sans Document
Calculatrice Interdite

1 Questions de cours

Les réponses doivent être courtes, quelques lignes suffisent [5]

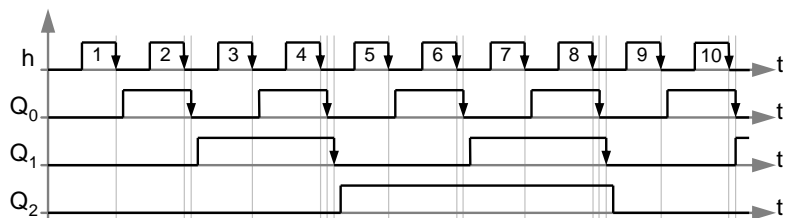
- 1.1 Comment peut-on construire un système séquentiel à partir d'un système combinatoire (vous pouvez faire un schéma) ? [1]
- 1.2 Donner un exemple de bascule asynchrone et de bascule synchrone. Expliquer le mode de fonctionnement de ces deux bascules. [1]
- 1.3 Expliquer les différences entre un compteur synchrone et un compteur asynchrone. Donner un schéma de chaque type de compteur. [1]
- 1.4 Appliquons un signal d'horloge de 512 kHz à une bascule dont les entrées J et K sont toutes les deux à 1. Quelle est la fréquence de l'onde de sortie de la bascule ? [1]
- 1.5 Donner les principales caractéristiques des mémoires SRAM et des mémoires DRAM. [1]

2 Exercice 1 [4]



Considérons le circuit de la figure, dont les entrées asynchrones sont supposées inactives.

- 2.1 Donner les chronogrammes des sorties des bascules pour dix impulsions d'horloge. [3]
- 2.2 Donner la fonctionnalité du circuit. [1]



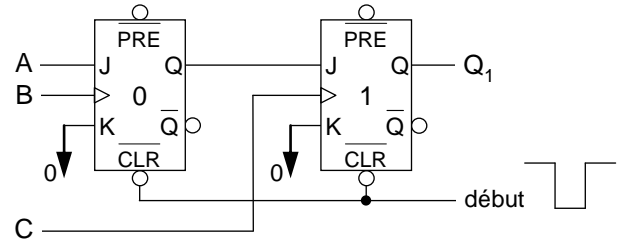
Il s'agit d'un compteur asynchrone série modulo 8, diviseur de fréquence par 2, 4 et 8.

3 Exercice 2 [3]

Considérons le schéma de la figure ci contre. L'entrée PRE est supposée inactive.

3.1 Expliquer le rôle de l'impulsion début. [1]

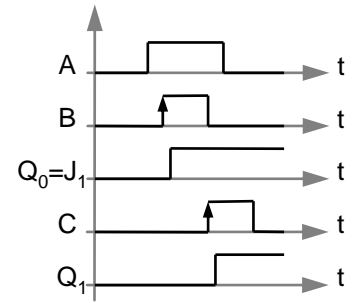
L'impulsion de début sert à effectuer l'initialisation du circuit en mettant à 0 toutes les sorties Q_i .



Les entrées A, B, C étant à zéro, la sortie Q_1 ne passera à 1 que quand A, B, C passeront à 1 en respectant une certaine séquence.

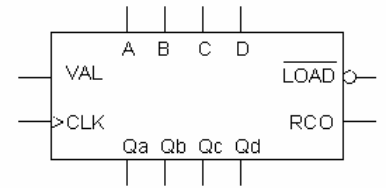
3.2 Donner la séquence de ces entrées qui fait passer Q_1 à 1. [2]

Remontons le temps. Q_1 passera à 1 avec un front montant de son horloge (C). K_1 étant à 0, il faut que $J_1 = Q_0$ passe à 1. Q_0 passera à 1 avec un front montant de son horloge (B). K_0 étant à 0, il faut que $J_0 = A$ passe à 1.



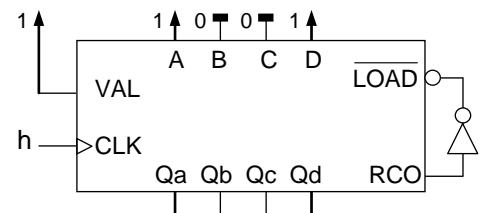
4 Exercice 3 [4]

Le circuit suivant est un compteur modulo 16. RCO est la sortie retenue. LOAD est une entrée autorisant le chargement parallèle des entrées A, B, C, D. L'entrée VAL est une entrée validation autorisant le fonctionnement du circuit.

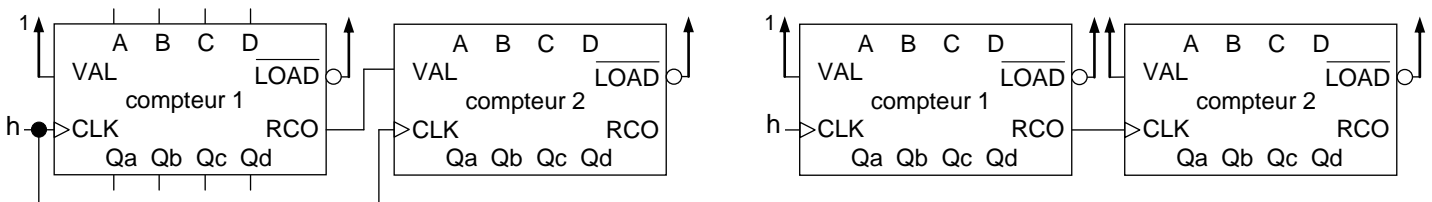


4.1 Proposer un schéma de montage réalisant un cycle d'initialisation de 0 à 15 puis des cycles de 9 à 15. [1]

À l'initialisation les entrées A, B, C, D et les sorties Q_a, Q_b, Q_c, Q_d, RCO sont à 0. Donc sur chaque front montant de h, le compteur compte de 0 à 15 où \overline{RCO} passe à 1 et $\overline{LOAD} = \overline{RCO}$ est active, chargeant $1001_2 = 9_{10}$ dans les entrées A, B, C et D compteur. Au coup d'horloge suivant le compteur compte à partir de 9_{10} jusqu'à 15_{10} etc.



4.2 Proposer un schéma de montage, utilisant deux compteurs réalisant un cycle de 0 à 255. [1]

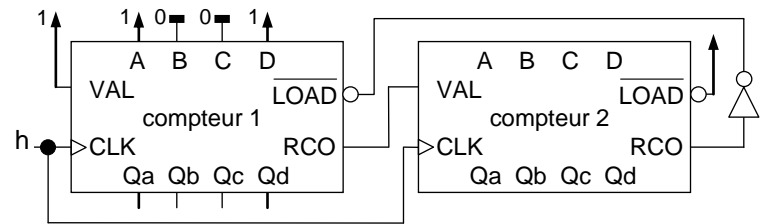


Compteur synchrone : la 16^{me} impulsion de h fait passer Cmpt1 à 15 et RCO_1 à 1 ce qui réveille Cmpt2 qui incrémente son compte à l'impulsion suivante, la 17^{me} , puis se remet en veille (RCO_1 étant repassée à 0)

Compteur asynchrone : les 2 compteurs sont toujours en marche. $RCO_1 = CLK_2$: le compteur Cmpt2 ne compte qu'à chaque fois que le compteur Cmpt1 passe à 15.

4.3 Proposer un schéma de montage, utilisant deux compteurs réalisant un cycle d'initialisation de 0 à 255 puis des cycles de 9 à 255. [2]

À l'initialisation les entrées A, B, C, D et les sorties Qa, Qb, Qc, Qd, RCO de Cmp₁ et de Cmp₂ sont à 0. Donc sur chaque front montant de h, Cmp₁ compte de 0 à 15 où RCO₁ passe à 1 et réveille Cmp₂ qui incrémente son compte à l'impulsion suivante, la 17^{me}. Au compte 15 de Cmp₂, RCO₂ passe à 1 et $\overline{LOAD_1} = \overline{RCO_2}$ active le chargement de $1001_2 = 9_{10}$ dans Cmp₁. Au coup d'horloge suivant Cmp₁ compte à partir de 9 jusqu'à 15₁₀ et C₂ repart de 0 etc.



5 Exercice 4 [4]

Nous souhaitons réaliser un plan mémoire comprenant : 16 Kilo de RAM à partir de l'adresse \$2000. Ce plan mémoire est adressable par un microprocesseur délivrant 16 bits d'adresse. Chaque mémoire RAM utilisée contient 4 Kilo Octets par boîtier.

5.1 Déterminer les espaces adressés par chaque RAM. [1,5]

Un boîtier : $4ko = 4 \times 2^{10} = 2^{12}$ mots de 8 bits, @ de 0 à 4095₁₀ ou de 0000 0000 0000₂ à 1111 1111 1111₂ ou de 000₁₆ à FFF₁₆. Il faut 4 boîtiers pour obtenir 16 ko.

Le µprocesseur @ : B₁ de \$2000 à \$2FFF, B₂ de \$3000 à \$3FFF, B₃ de \$4000 à \$4FFF, B₄ de \$5000 à \$5FFF.

5.2 Proposer un schéma réalisant ce décodage en utilisant un décodeur à 3 entrées. [2,5]

Nous prenons les 3 bits d'@ de poids 12, 13 et 14 pour commander les @ a₀, a₁ et a₂ du décodeur 3x8. Le bit d'@ A₁₅ commande la sélection \overline{CE} du décodeur afin que les @ \$A000 à \$DFFF ne soient pas envoyées vers ce plan mémoire.

Pour B₁ : $a_{14} a_{13} a_{12} = 010_2 = 2_{10}$
d'où $\overline{CS_1} = \overline{O_2}$

Pour B₂ : $a_{14} a_{13} a_{12} = 011_2 = 3_{10}$
d'où $\overline{CS_2} = \overline{O_3}$

Pour B₃ : $a_{14} a_{13} a_{12} = 100_2 = 4_{10}$
d'où $\overline{CS_3} = \overline{O_4}$

Pour B₄ : $a_{14} a_{13} a_{12} = 101_2 = 5_{10}$
d'où $\overline{CS_4} = \overline{O_5}$

