ES102/PC4 : énoncé

1) Portes CMOS complexes pour afficheur 7 segments et quelques leçons à en tirer

Malgré l'omniprésence d'écrans à tout faire, l'affichage de chiffres décimaux se contente encore souvent d'afficheurs plus simples, dits à 7 segments (tous allumés pour afficher le chiffre 8 : voir ci-contre). Pour les premières questions ci-dessous, on affiche en fait seulement les chiffres *k* entre 0 et 7 (version octale). Le code binaire de *k* est noté $(qdu)_2$. A chacun des 7 segments correspond donc une fonction booléenne de q, d et u.

1a) L'une d'elles est-elle croissante ? Ou décroissante ?

1b) Concevoir en logique complémentaire la porte CMOS qui fournit la fonction booléenne *sdb* servant à commander le **s**egment situé sur le côté **d**roit, en **b**as.

1c) Concevoir la porte CMOS fournissant la commande *sdh* du **s**egment situé côté **d**roit, en **h**aut.

1d) Dans le bloc *n* de la porte CMOS que l'on vient de synthétiser pour *sdh*, il est naturel de mettre en commun les transistors commandés par q. Quel en est le sens algébrique ?

1e) Simplifier le montage précédent dans le cas où q=1. Qu'obtient on ? Comment modifier le montage pour obtenir une porte XOR ?

1f) Comment se positionne le délai d'une porte XOR ainsi réalisée par rapport à celui de ses consoeurs ? Est-ce cohérent avec les performances de la *bibliothèque* de portes de la PC2 ?

1g) Retrouver la FDM de *sdh* par la méthode de Quine.

On souhaite désormais afficher les 10 chiffres décimaux, d'où *k* codé sur 4 bits : $(hqdu)_2$. Cependant, les possibilités 10 à 15 sont inexploitées. A chacun des 7 segments de l'afficheur correspond donc une fonction booléenne *incomplètement spécifiée*.

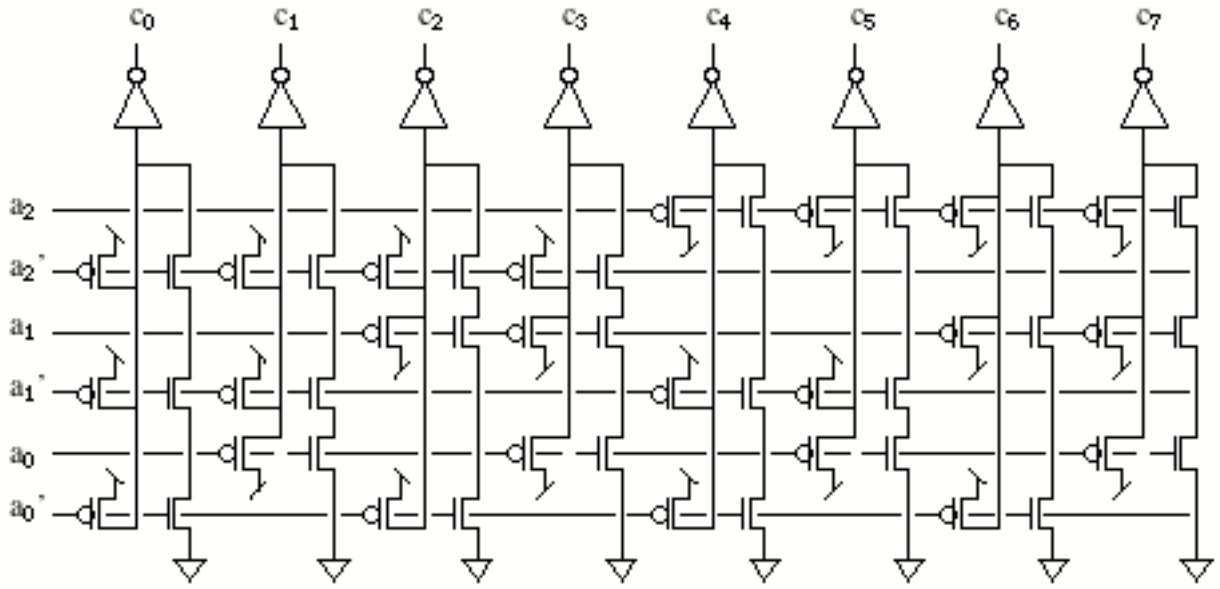
1h) Traitant d'abord les cas indifférents comme des 1, déterminer, sans écrire la moindre formule booléenne, le nombre de transistors nécessaires pour réaliser la porte CMOS fournissant la commande *sgh* du **s**egment situé sur le côté **g**auche, en **h**aut.

1i) Les cas indifférents peuvent-ils être mieux exploités pour minimiser le coût en transistors ?

1j) Au vu des deux questions précédentes, il serait tentant d'exploiter différemment les cas indifférents selon que l'on synthétise le bloc *p* ou le bloc *n*. Pourquoi ne faut-il pas le faire ?

1k) En fait, on reste sur des 1 pour les cas indifférents car cela va permettre une factorisation avantageuse de FDM. Achever de concevoir la porte CMOS fournissant *sgh*.

1l) Pour h=0, quel genre de fonction est *sgh* ? Dans l'implantation correspondante, constater que les formules structurelles des blocs *n* et *p* ne sont pas duales, contrairement aux observations faites jusque-là sur des cas plus simples. Si on forçait γ_n à être duale de γ_p, que se passerait-il ? Quelle conclusion en tirer ?

2) Question d'adresse

Quelle est la fonction réalisée par la structure CMOS ci-dessous ?