

L'examen est composé de deux parties indépendantes.

Partie 1 : Électronique analogique

Exercice 1 - Amplification

On considère l'amplificateur de la Figure 1 gauche). Le transistor nMOS N1 fonctionne en source de courant avec un courant non nul $I_{ds1} > 0$. On indicera respectivement 1 les grandeurs relatives à N1 et 2 les grandeurs relatives au transistor pMOS P2. On rappelle que les bornes G, D et S désignent respectivement la grille, le drain et la source des transistors.

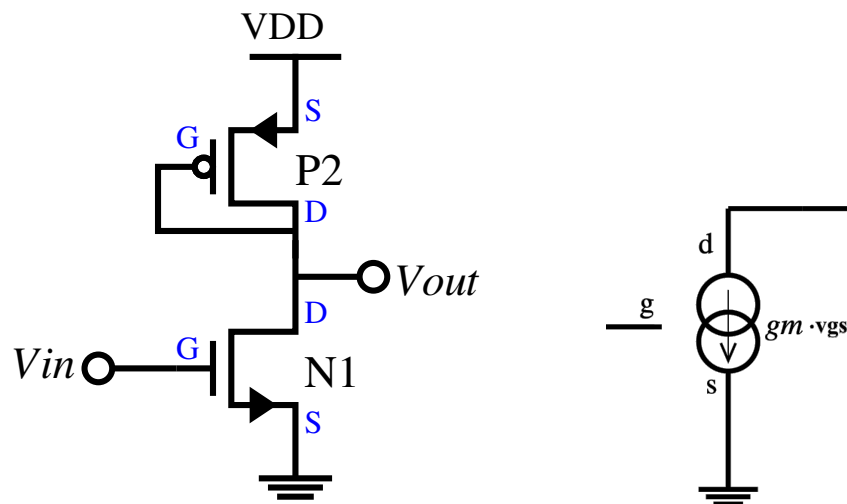


FIGURE 1 – Gauche) Architecture de l'amplificateur Droite) Modèle du transistor fonctionnant en source de courant, linéarisé autour d'un point de fonctionnement

Question 1.1 Quelles sont les deux inégalités vérifiées par les tensions V_{ds1} , V_{T1} et V_{gs1} caractérisant les zones de fonctionnement du transistor N1 ?

- Réponse 1.1** — Interrupteur ouvert pour $V_{gs} < V_T$ où V_T est la tension de seuil du transistor
- Source de courant pour $(V_{gs} > V_T) \ \& \ (V_{ds} > V_{gs} - V_T)$
 - Résistance pour $(V_{gs} > V_T) \ \& \ (V_{ds} < V_{gs} - V_T)$

Le transistor P2 est un transistor pMOS dont la source est connectée à V_{DD} , le drain et la grille sont connectés à V_{out} . Les tensions V_{GS2} , V_{T2} et V_{DS2} ainsi que le courant I_{ds2} sont négatifs. En prenant les valeurs absolues de toutes ces grandeurs, les relations régissant le fonctionnement du pMOS sont identiques à celles du nMOS.

Question 1.2 Démontrer que le transistor P2 fonctionne également en source de courant.

Réponse 1.2 Comme $V_{gs2} = v_{ds2}$, l'inégalité $|V_{ds2}| > |V_{gs2}| - |V_T|$ est forcément respectée, le transistor pMOS est donc soit bloqué soit en source de courant mais $I_{DS1} = -I_{SD2} > 0$, le transistor P2 est donc forcément en zone source de courant.

Le schéma équivalent linéaire ou petit signal des 2 transistors est identique et donné sur la Figure 1 droite). Il est constitué d'une source de courant commandée par v_{gs} .

Question 1.3 Démontrer que le modèle petit signal du transistor pMOS P2 dans cette configuration est une résistance. Déterminer son expression.

Réponse 1.3 Le courant circulant entre le drain et la source du transistor est égal à $i_{ds2} = g_m \cdot v_{gs2} = g_{m2} \cdot v_{ds2}$. Donc le transistor agit une comme résistance de valeur $1/g_{m2}$

Question 1.4 Elaborer le modèle linéaire de l'amplificateur de la Figure 1 gauche).

Réponse 1.4 Le modèle est simplement une source de courant représentant le transistor nMOS et d'une résistance $1/g_{m2}$ pour le transistor pMOS.

Question 1.5 Etablir l'expression du gain linéaire $h = \frac{v_{out}}{v_{in}}$ de l'amplificateur en fonction de g_{m1} et g_{m2} .

Réponse 1.5 Le gain est donc $h = \frac{g_{m1}}{g_{m2}}$

Exercice 2 - Etude d'une cellule du second ordre en capacités commutés

L'objectif de cet exercice est de trouver la fonction de transfert de la cellule du second ordre de la Figure 2. Pour cela, nous allons étudier comment remplacer certaines structures de circuit par leur schéma bloc fonctionnel.

On considère le circuit à capacités commutées de la Figure 3. La sortie est échantillonnée à la phase ϕ_2 . L'amplificateur opérationnel, les commutateurs et les condensateurs sont parfaits (les transferts de charges sont instantanés). L'entrée V_i est bloquée sur la phase 2 (ϕ_2). Ceci se traduit par $V_i^{\phi_1}((n-1/2)T_e) = V_i^{\phi_2}((n-1)T_e)$ ($i \in 1, 2, 3$).

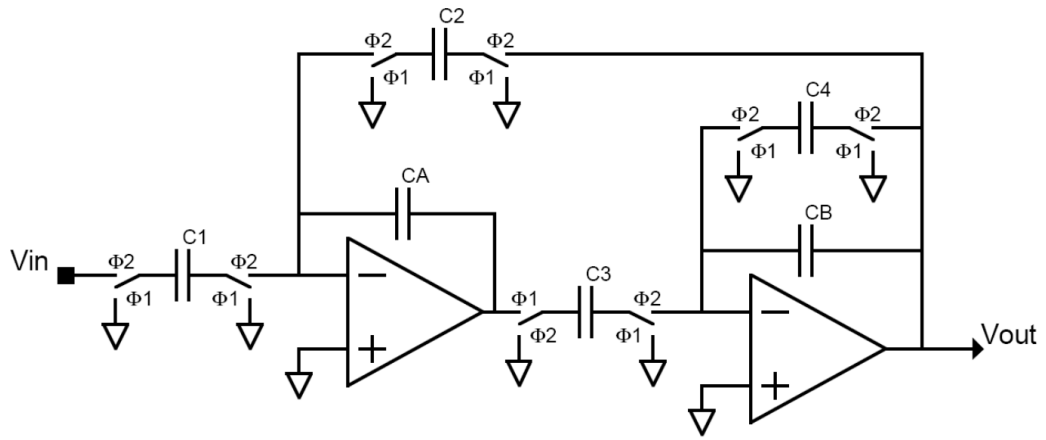


FIGURE 2 – Cellule du second ordre à étudier

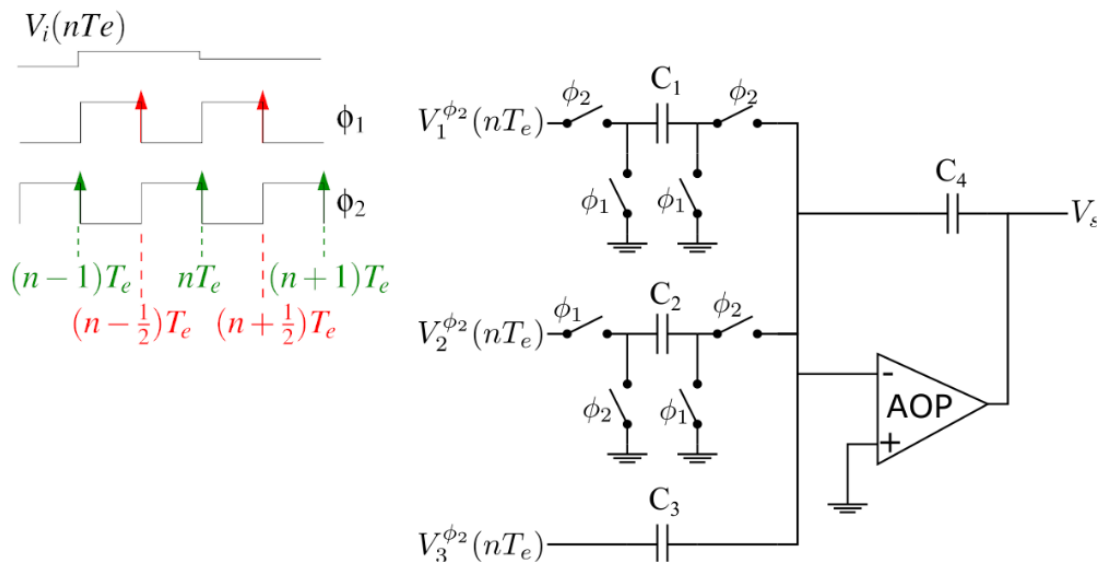


FIGURE 3 – Circuit à capacités commutées

Question 2.1 Déterminer la relation de récurrence qui relie $V_s^{\phi_2}(nT_e)$ à $V_1^{\phi_2}(nT_e)$, $V_2^{\phi_2}(nT_e)$ et $V_3^{\phi_2}(nT_e)$.

Réponse 2.1 On calcule les charges aux instants pairs et impairs puis la conservation de charge. En phase P, la conservation de charge concerne tous les condensateurs, en phase I uniquement C_3 et C_4 . On obtient la fonction suivante :

$$V_s(nT_e) = -\frac{C_1}{C_4}V_1(nT_e) + \frac{C_2}{C_4}V_2(nT_e - T_e) - \frac{C_3}{C_4}(V_3(nT_e) - V_3(nT_e - T_e)) + V_s(nT_e - T_e)$$

On fait la transformée en Z, on obtient

$$V_s(z) = \frac{1}{C_4(1 - z^{-1})}(-C_1V_1(z) + C_2V_2(z)z^{-1} - C_3(1 - z^{-1})V_3(z))$$

On considère le bloc fonctionnel de la Figure 4.

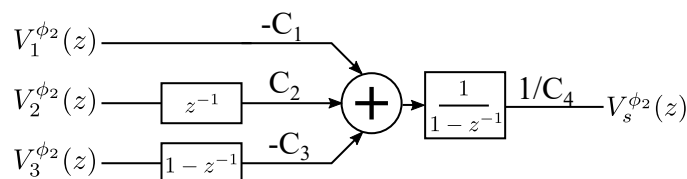


FIGURE 4 – Bloc fonctionnel

Question 2.2 Déterminer la relation qui relie $V_s^{\phi_2}(z)$ en fonction de $V_1^{\phi_2}(z)$, $V_2^{\phi_2}(z)$ et $V_3^{\phi_2}(z)$. La comparer avec la transformée en Z de la relation trouvée à la question précédente.

Réponse 2.2 On obtient la même fonction de transfert, ce qui montre que le bloc fonctionnel est équivalent au circuit à capacités commutées.

Nous venons de voir qu'il est possible de remplacer certaines structures d'un circuit à capacités commutées par un schéma bloc équivalent. Ainsi chacune des branches d'entrée du circuit de la Figure 3 peut être remplacée par celle de la Figure 4. De même l'amplificateur avec rétroaction capacitive sur l'entrée négative est équivalent au bloc fonctionnel après le sommateur.

Question 2.3 Déterminer le schéma bloc fonctionnel du circuit de la Figure 2.

Réponse 2.3

Question 2.4 Démontrer que la fonction de transfert en z du filtre est donnée par la relation suivante :

$$T(z) = -\frac{C_1 C_3}{C_A C_B} \cdot \frac{z}{\left(\frac{C_A}{C_B} + 1\right) z^2 + \left(\frac{C_2 C_3}{C_A C_B} - \frac{C_4}{C_B} - 2\right) z + 1}$$

Réponse 2.4 En utilisant le schéma, on retrouve cette fonction.

En fixant $C_1 = C_2 = C_3 = 1$ pF, $C_4 = 0.1$ pF, $C_A = C_B = 3.18$ pF et la fréquence d'échantillonnage $f_e = 100$ MHz, la fonction réalisée par le circuit étudié est un filtre passe bande centré autour de $f_0 = 5$ MHz, avec un facteur de qualité Q de 10 et un gain à f_0 de 20 dB.

Question 2.5 Tracer à main levée, le module du filtre entre 0 et 100 MHz.

Réponse 2.5 Filtre passe bande avec une symétrie autour de 50 MHz.

Partie 2 : Électronique numérique

Exercice 3 - Mise en œuvre d'un opérateur de calcul séquentiel

La figure 5 représente l'architecture d'un opérateur de calcul séquentiel.

Cet opérateur a les caractéristiques suivantes :

- les deux entrées **X** et **Y** sont deux nombres positifs représentés sur n bits,
- la sortie **A**, aussi un nombre positif encodé sur m bits,

Les registres utilisent l'horloge **clk** et possèdent une entrée de remise à zéro synchrone **nrst** active sur niveau bas. Ces deux signaux ne sont pas représentés dans le schéma pour ne pas l'encombrer.

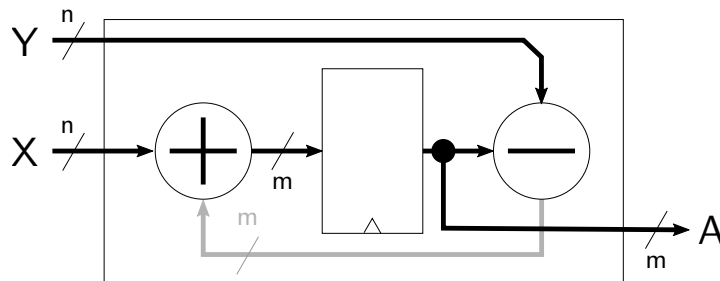


FIGURE 5 – Architecture d'un opérateur séquentiel

Les entrées viennent elles aussi de registres utilisant la même horloge.

Question 3.1 Pour $n=8$ et $m=10$ complétez le code SystemVerilog suivant pour décrire cet opérateur.

Réponse 3.1

```
module ope (
    input logic clk,
    input logic nrst,
    input logic [7:0] X,
    input logic [7:0] Y,
    output logic [9:0] A
);
always_ff@(posedge clk)
if(!nrst)
    A <= 0;
else
    A <= A - Y + X;
endmodule
```

```

module ope (...
    input logic [...] X,
    input logic [...] Y,
    ...
);
...
endmodule

```

Question 3.2 Donnez l'expression *mathématique* permettant de calculer la valeur de la sortie **A** à un l'instant **k** ($A_k = \dots$) en fonction des entrées et d'éventuels états précédents du système.

Réponse 3.2

$$A_{k+1} = (A_k - Y_k + X_k) \bmod 2^m$$

Question 3.3

- Donnez la définition du chemin critique dans un système séquentiel synchrone.
- En considérant que les entrées **X** et **Y** viennent directement de la sortie de registres, situez le chemin critique dans l'opérateur de la figure 5.

Réponse 3.3

Le chemin **cominatoire** avec le temps de propagation le plus long.

Pour l'opérateur, il se situe de la sortie du registre **A** à l'entrée du registre **A**, ou de l'entrée **Y** à l'entrée du registre **A** ou encore à travers le soustracteur suivi de l'additionneur.

L'additionneur et le soustracteur de la figure 5 ont une structure standard (dite à propagation de retenue) formée d'additionneurs 1 bit chaînés.

Pour chaque additionneur 1 bit, nous connaissons les temps de propagation :

- entre les entrées et la sortie *somme* t_s ,
- entre les entrées et la sortie *retenue* t_c ,

Question 3.4 Exprimez la fréquence maximale de fonctionnement de l'opérateur de la figure 5 en fonction de t_s , t_c et de la taille des entrées et sorties (n et m).

Réponse 3.4

$$\sim \frac{1}{m \times t_c}$$

le temps de propagation est \simeq le temps de propagation d'un additionneur. ou

$$\frac{1}{t_s + m \times t_c}$$

le temps de propagation est \simeq le temps de propagation d'un additionneur + une couche d'addition ou si on prend en compte la propagation et les contraintes sur les registres

$$\frac{1}{t_s + m \times t_c + t_{co} + t_{su}}$$

Est aussi accepté les variantes avec $t_{add/sub} = (m - 1) \times t_c + \max(t_s, t_c)$

L'opérateur de la figure 5 produit un résultat à chaque front d'horloge. Nous voulons ajouter une entrée d'activation (**enable**), telle que l'opérateur ne produise un nouveau résultat que si cette entrée est à l'état haut.

Question 3.5 Modifiez le code SystemVerilog précédent pour ajouter ce comportement.

Réponse 3.5

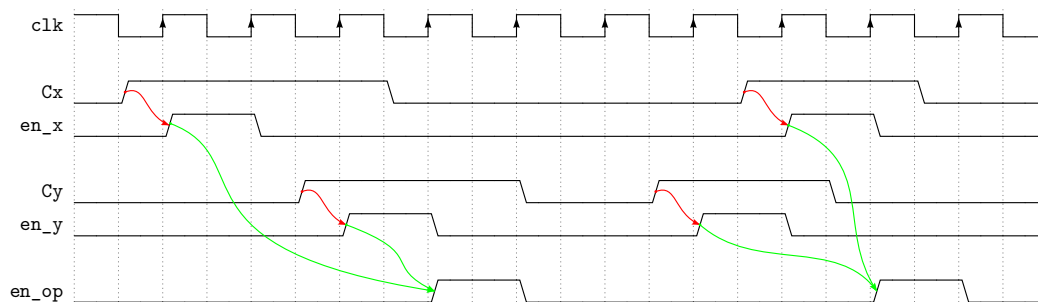
```

module ope (
    input logic clk,
    input logic nrst,
    input logic enable,
    input logic [7:0] X,
    input logic [7:0] Y,
    output logic [9:0] A
);
always_ff@(posedge clk)
if(!nrst)
    A <= 0;
else if(enable)
    A <= A - Y + X;
endmodule

```

Des signaux annonçant l'arrivée des données **X** et **Y** viennent du monde extérieur (non synchrones). Nous voulons contrôler la capture de ces données et l'activation de l'opérateur de calcul en construisant un automate.

La figure suivante montre le comportement voulu.



- en passant à 1, **Cx** (resp. **Cy**) demande à capturer **X** (resp. **Y**), ce signal peut durer plusieurs périodes d'horloge, il faut générer un signal **en_x** (resp. **en_y**) d'une durée d'exactly un cycle d'horloge pour autoriser la capture.
- **Cx** et **Cy** peuvent arriver simultanément ou décalés dans le temps, dans un ordre quelconque, mais, nous avons la garantie que les deux sont descendus à 0, avant que l'un puisse repasser à 1.
- une fois que les deux valeurs capturées, il faut activer l'opérateur en mettant à 1 le signal **en_op** durant un cycle.

Question 3.6


```

    if(Cx && !Cy) state <= EX1;
    if(!Cx && Cy) state <= EY1;
    if(Cx && Cy) state <= EXY;
end
EX1: if(Cy) state <= EY2; else state <= WY;
WY : if(Cy) state <= EY2;
EY2: state <= Eop;
EY1: if(Cx) state <= EX2; else state <= WX;
WX : if(Cx) state <= EX2;
EX2: state <= Eop;
EXY: state <= Eop;
Eop: state <= W;
W : if (!Cx && !Cy) state <= INIT;
endcase

```

```

assign en_x = (state == EX1) || (state == EX2) || (state == EXY);
assign en_y = (state == EY1) || (state == EY2) || (state == EXY);
assign en_op = (state == Eop);
endmodule

```

```

module ctrl (...
    input  logic Cx, Cy,
    output logic en_x, en_y, en_op
);

    enum logic [...] {INIT,...} state;
    ...
endmodule

```