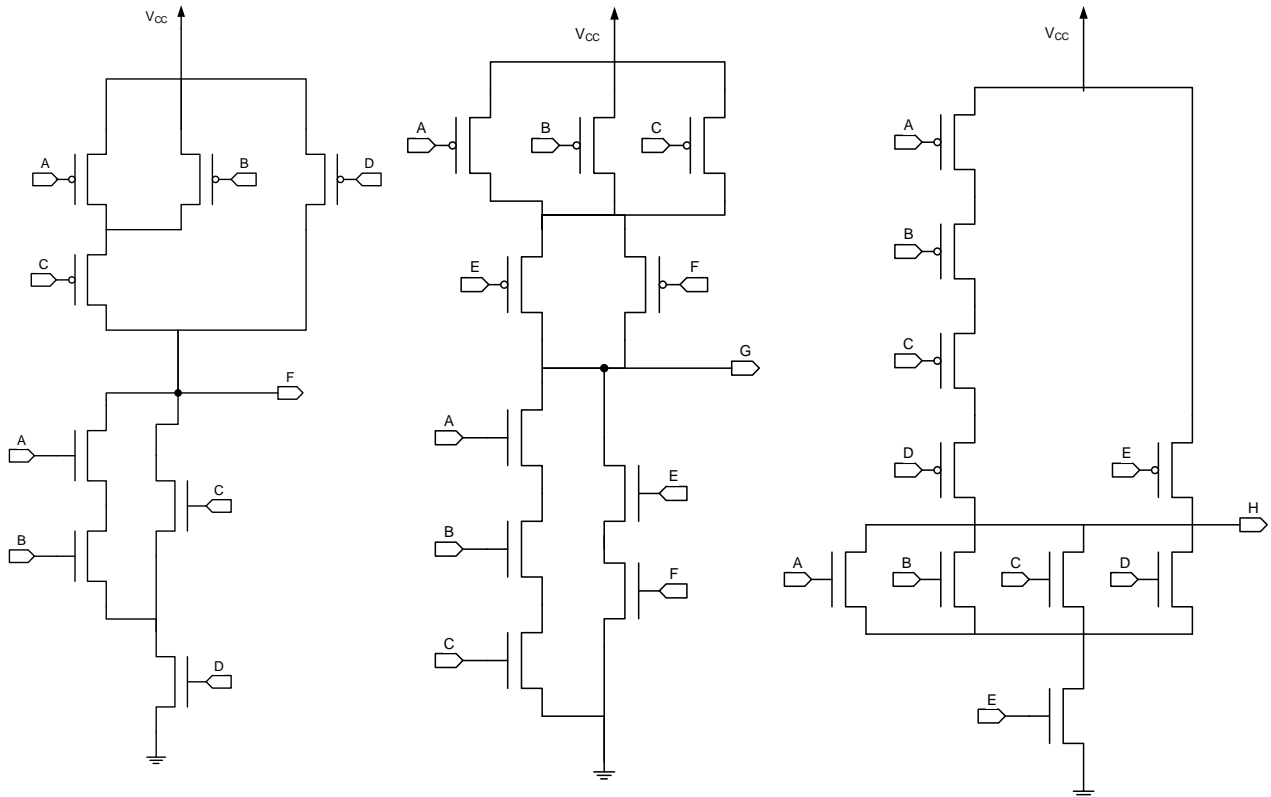


1. Donnez la fonction logique réalisée par les circuits CMOS suivants.



2. Donnez un circuit CMOS pour réaliser les fonctions logiques de la table de vérité suivante.

Les entrées A, B et C ne sont pas disponibles en version inversée.

Utilisez le moins de transistors possible.

A	B	C	T	U	V	W	X
0	0	0	1	0	1	1	0
0	0	1	1	0	1	1	0
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	0
1	0	0	1	0	1	0	1
1	0	1	0	0	1	1	1
1	1	0	1	0	1	1	1
1	1	1	0	0	1	1	1

### 3-Mémoires PROM

3. Estimez le nombre de transistors, de diodes et de fusibles programmables utilisés dans une ROM à 16 mots de 8 bits si elle est réalisée avec un circuit CMOS.

4. Programmer le PROM suivant pour implémenter le code VHDL qui suit. Étiqueter correctement les ports d'entrée et de sortie, et placer des boulets (•) aux intersections de lignes à relier ensemble.

```

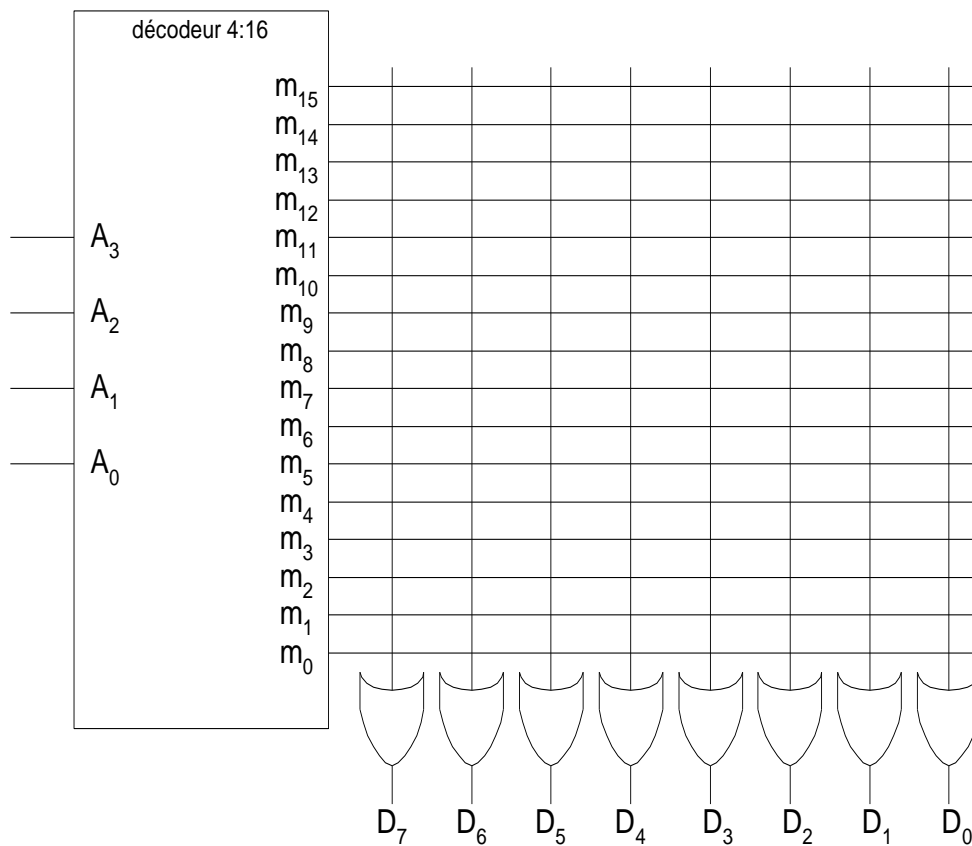
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;

-- multiplie le nombre en entrée par 3

entity fois3 is
port (
    n : in unsigned(2 downto 0);
    nfois3: out unsigned(4 downto 0)
);
end fois3;

architecture arch1 of fois3 is
signal temp : natural range 0 to 21;
begin
    with to_integer(n) select
    temp <=
        0 when 0,
        3 when 1,
        6 when 2,
        9 when 3,
        12 when 4,
        15 when 5,
        18 when 6,
        21 when 7,
        0 when others;
    nfois3 <= to_unsigned(temp, 5);
end arch1;

```



### 5-PLA, PAL, GAL et CPLD

5. Vrai ou faux : une mémoire PROM est moins efficace, en termes du pourcentage du circuit utilisé, qu'un circuit PLA pour implémenter la plupart des fonctions logiques. Justifiez votre réponse.

6. Soit le code VHDL suivant. Compléter le circuit PAL donné pour implémenter la fonctionnalité du module `combinatoire2`. Étiqueter correctement les ports d'entrée et de sortie, et placer des boulets (•) aux intersections des lignes à relier ensemble.

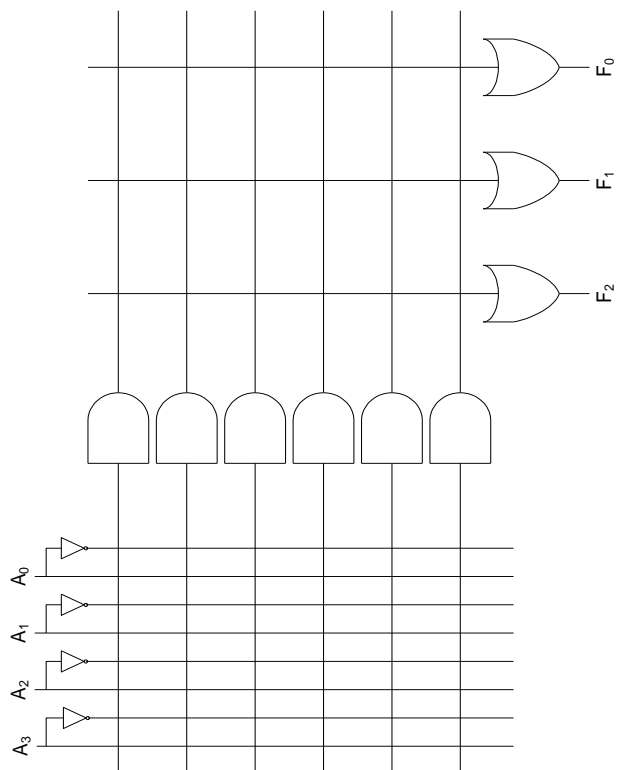
```

library ieee;
use ieee.std_logic_1164.all;

entity combinatoire2 is
  port (
    x1, x2, x3 : in std_logic;
    y1, y2 : out std_logic
  );
end combinatoire2;

architecture arch of combinatoire2 is
  signal p3, p4 : std_logic;
begin
  y2 <= p3 or p4;
  p3 <= not x1 and not x2;
  p4 <= x1 and not x2 and not x3;
  y1 <= x1 xor x3;
end arch;

```



### 6-Les FPGA: structure, fonctionnement et utilisation

7. À propos des FPGA

- Donnez le diagramme général d'un FPGA; identifiez et décrivez brièvement les trois types de composants principales.
- Donnez le diagramme général d'une tranche de type L d'un FPGA Artix 7 et identifiez les cinq composants principales. Il n'est pas nécessaire de montrer toutes les interconnexions possibles.
- Énumérez et décrivez brièvement cinq composants secondaires d'un FPGA.

8. Implémentez le module décrit par le code VHDL suivant avec un circuit CMOS, la ROM donnée, le PLA donné, le PAL donné et le FPGA donné.

Étiquetez correctement les ports d'entrée et de sortie. Pour la ROM, le PLA et le PAL, placez des boulets (•) aux intersections de lignes à relier ensemble.

Pour le FPGA, indiquer directement sur le dessin où chaque signal se situe ainsi que les interconnexions entre les blocs. Indiquer dans les tables de vérité fournies le contenu de chacune des tables de conversion que vous utilisez.

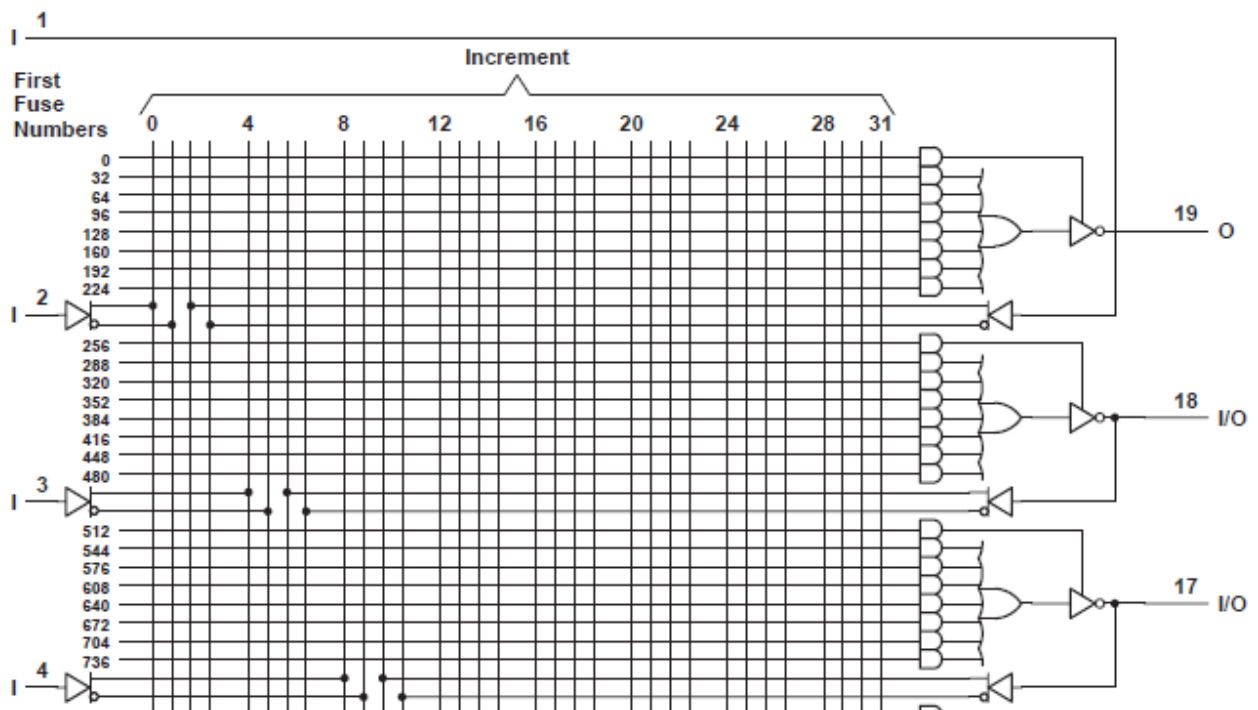
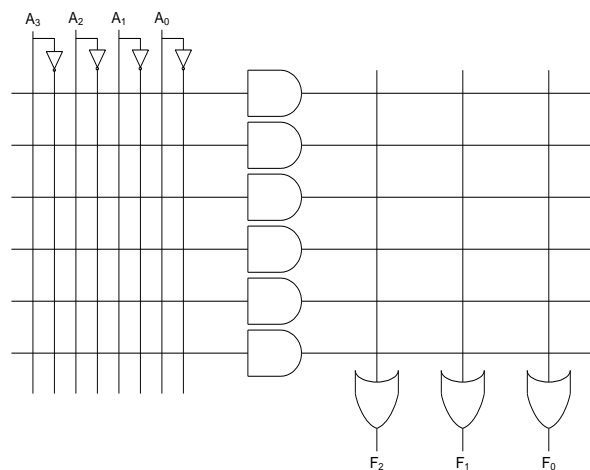
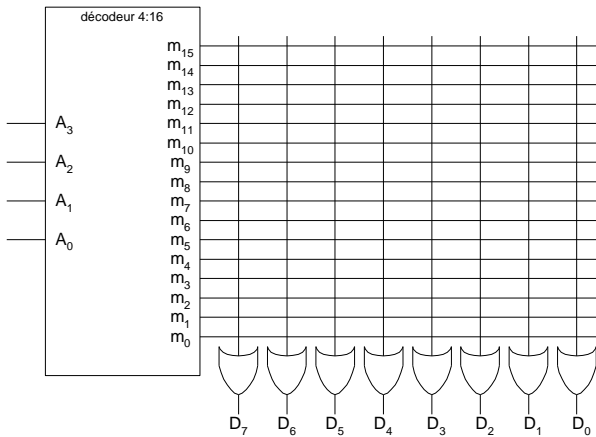
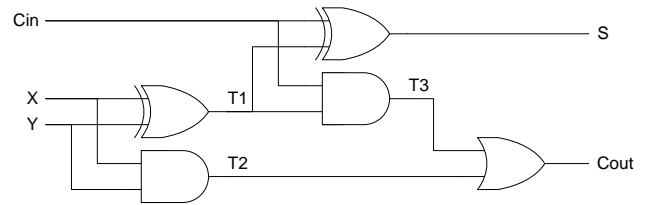
```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity add3bits is
  port (
    Cin, X, Y : in std_logic;
    Cout, S : out std_logic
  );
end add3bits;

architecture flotdonnees of add3bits is
  signal T1, T2, T3 : std_logic;
begin
  S <= T1 xor Cin;
  Cout <= T3 or T2;
  T1 <= X xor Y;
  T2 <= X and Y;
  T3 <= Cin and T1;
end flotdonnees;

```



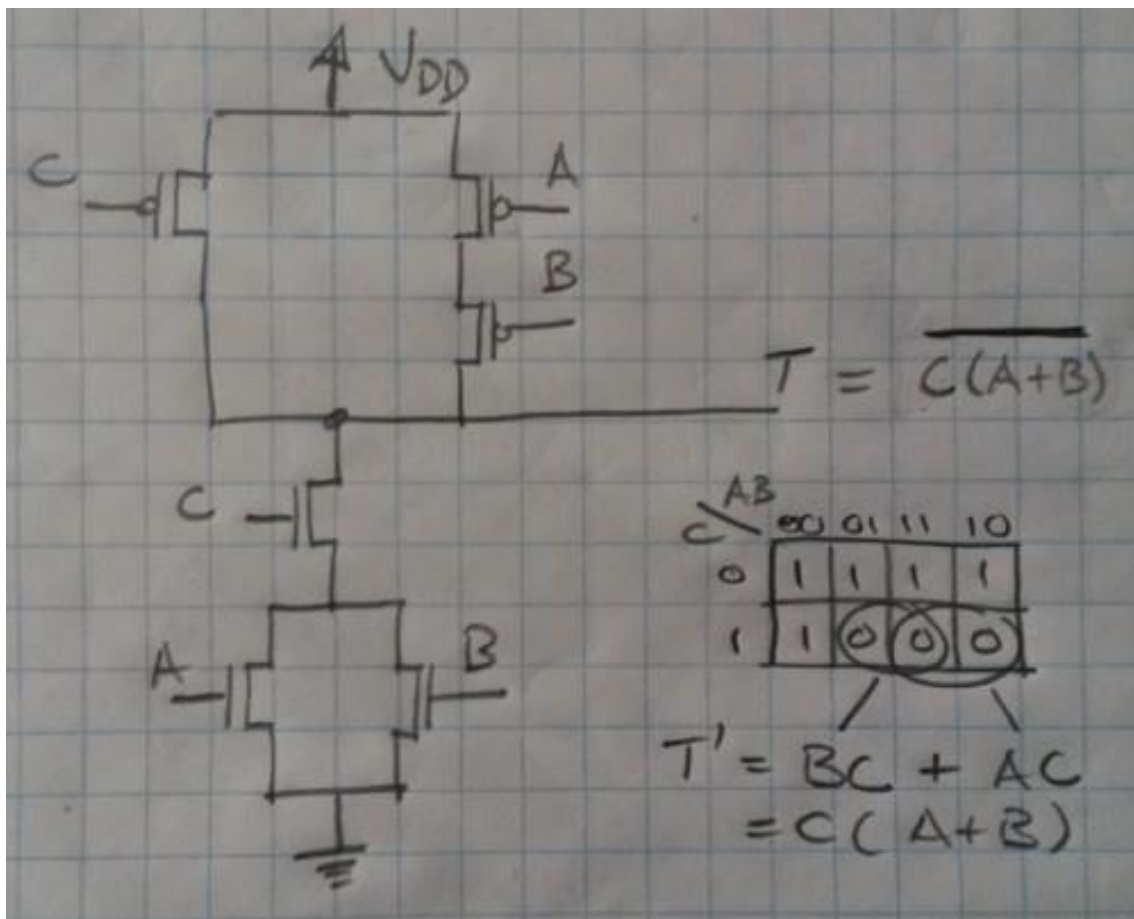


Solutions

1. Réponses :  $F = ((AB + C) \cdot D)'$ ;  $G = (ABC + EF)'$ ;  $H = ((A + B + C + D) \cdot E)'$

2. Réponse

On a  $T = A'B' + C' = [(A + B)C]'$ . Il faut donc 6 transistors. Le réseau NMOS a A et B en parallèle, avec la paire en série avec C. Le réseau PMOS a C en parallèle avec la paire A et B qui est en série.



Voici un circuit pour W. Il faut ajouter 2 inverseurs, donc 4 transistors, pour générer B' à partir de B et C' à partir de C.

