



# Master 2-SEM

## Concepts avancés d'Architecture

**TD 1 - VHDL**

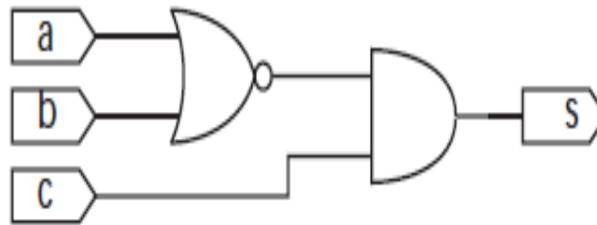
*Année 2022-2023*

*Pr. R. BOUDOUR*

# QCM

2

Question 1. Quelles sont les expressions logiques correspondant au logigramme ci-dessous ?



(a)  $\neg a + b \cdot c$

(b)  $\neg a + \neg b \cdot c$

(c)  $\neg(a + b) \cdot c$

(d)  $(\neg a + \neg b) \cdot c$

(e)  $\neg a \cdot \neg b \cdot c$

(f)  $\neg(a \cdot b) \cdot c$

# QCM

3

**Question 2.** Parmi les expressions logiques ci-dessous, quelles sont celles qui donnent pour résultat l'état logique de  $a$  ?

(a)  $a . a$

(b)  $a + a$

(c)  $a + 1$

(d)  $a . 1$

(e)  $a + 0$

(f)  $a . 0$

(g)  $a + /a$

(h)  $a . /a$

(i)  $a . (b + /b)$

(j)  $b . (a + /a)$

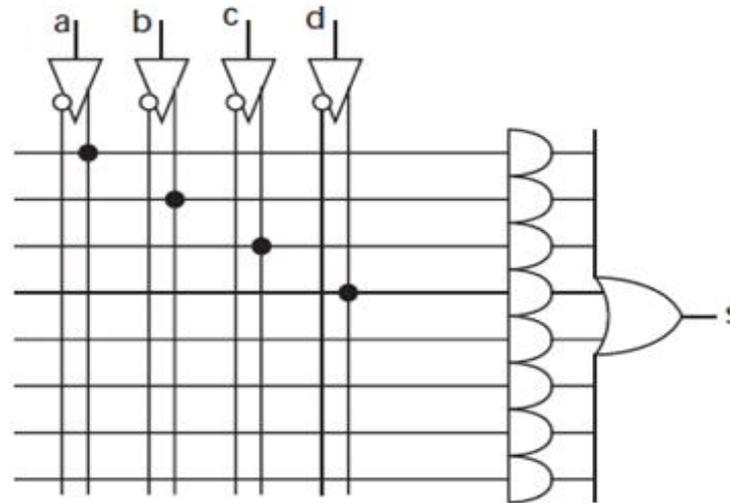
(k)  $a + (/a . b . /b)$

(l)  $/a$

# QCM

4

**Question 3.** A quelles équations correspond le PLD programmé suivant ?



(a)  $a . b . c . d$

(b)  $\overline{a . b . c . d}$

(c)  $\overline{a + b + c + d}$

(d)  $\overline{a . b . c . d}$

(f)  $\overline{a + b + c + d}$

(g)  $a + b + c + d$

(h) 1

(i) 0

# QCM

5

Question 4. A quelles équations correspond la table de vérité suivante ?

a	b	c	d	s
0	X	X	X	1
1	0	X	X	1
1	1	0	X	1
1	1	1	0	1
Autres				0

(a)  $a \cdot /b \cdot /c \cdot /d + a \cdot b \cdot /c \cdot /d + a \cdot b \cdot c \cdot /d$

(b)  $a \cdot b \cdot c \cdot /d$

(c)  $/a + a \cdot /b + a \cdot b \cdot /c + a \cdot b \cdot c \cdot /d$

(d)  $a \cdot /b \cdot /c \cdot /d$

(e) 0

(f) 1

# QCM

6

**Question 5.** Comment s'écrit en VHDL l'expression  $s = a . b$  ?

(a)  $s = a \&\& b;$

(b)  $s = a \& b;$

(c)  $s = a * b;$

(d)  $s = a \text{ AND } b;$

(e)  $s \leq a \&\& b;$

(f)  $s \leq a \& b;$

(g)  $s \leq a * b;$

(h)  $s \leq a \text{ AND } b;$

**Question 6.** En VHDL, comment se déclare en sortie du circuit un signal  $s$  de type logique ?

(a) `OUT s : STD_LOGIC;`

(b) `STD_LOGIC s : OUT;`

(c) `STD_LOGIC OUT : s;`

(d) `OUT STD_LOGIC : s;`

(e) `s : OUT STD_LOGIC;`

(f) `STD_LOGIC : s OUT;`

# QCM

8

## Question 7.

**Question 2.** On souhaite décrire en VHDL une fonction logique dont la table de vérité est donnée ci-dessous. Quelles descriptions VHDL correspondent à cette fonction ?

abc	s
xx0	0
001	0
101	1
011	1
111	1

(a) 

```
ENTITY myFunc IS
  PORT (
    a, b, c: IN std_logic;
    s: OUT std_logic
  );
END myFunc;

ARCHITECTURE ar OF myFunc IS
BEGIN
  s <= a OR (b AND c);
END ar;
```

(b) 

```
ENTITY myFunc IS
  PORT (
    a, b, c: IN std_logic;
    s: OUT std_logic
  );
END myFunc;

ARCHITECTURE ar OF myFunc IS
BEGIN
  s <= c AND (a OR b);
END ar;
```

(c) 

```
ENTITY myFunc IS
  PORT (
    a, b, c: IN std_logic;
    s: OUT std_logic
  );
END myFunc;

ARCHITECTURE ar OF myFunc IS
  SIGNAL abc: std_logic_vector(2 DOWNTO 0);
BEGIN
  abc <= a & b & c;
  WITH abc SELECT
    s <=
      '1' WHEN "011" | "101" | "111",
      '0' WHEN OTHERS;
END ar;
```

(d) 

```
ENTITY myFunc IS
  PORT (
    a, b, c: IN std_logic;
    s: OUT std_logic
  );
END myFunc;

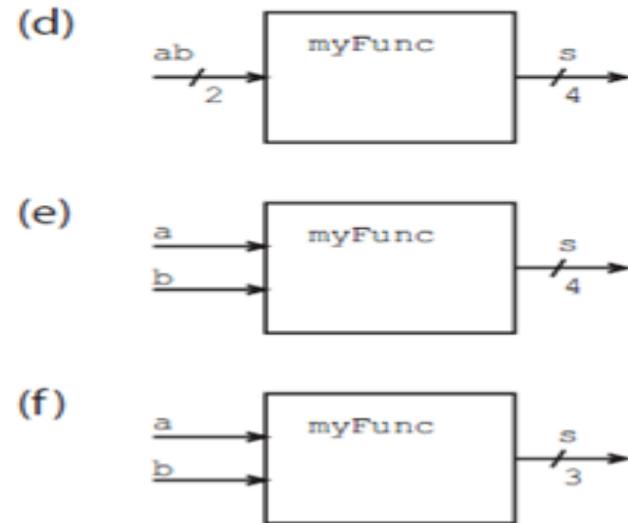
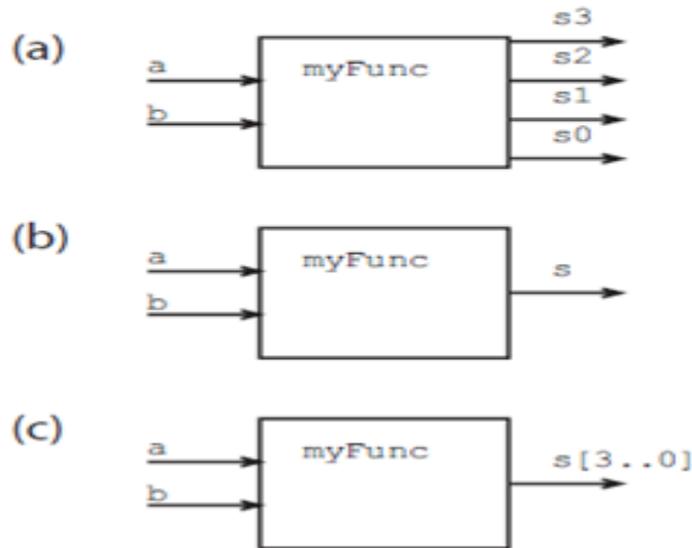
ARCHITECTURE ar OF myFunc IS
  SIGNAL ab : std_logic;
BEGIN
  s <= ab AND c;
  ab <= a OR b;
END ar;
```

# QCM

9

**Question 8 :** A quels schémas structurels correspond la déclaration VHDL ci-dessous ?

```
ENTITY myFunc IS
  PORT (
    a, b: IN std_logic;
    s: OUT std_logic_vector(3 downto 0)
  );
END myFunc;
```



# QCM

10

**Question 9 :** Parmi les propositions ci-dessous qui se rapportent à description d'un décodeur lesquelles sont vraies ?

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY decoder IS
  PORT (
    a: IN std_logic_vector(2 DOWNTO 0);
    b: OUT std_logic_vector(5 DOWNTO 0)
  );
END decoder;

ARCHITECTURE ar OF decoder IS
BEGIN
  WITH a SELECT
    b <-
      "000001" WHEN "000",
      "000010" WHEN "001",
      "000100" WHEN "010",
      "001000" WHEN "011",
      "010000" WHEN "100",
      "100000" WHEN "101",
      "000000" WHEN OTHERS;
END ar;
```

- ( a ) Le signal *a* représente l'adresse de la sortie active.
- ( b ) Le signal *a* représente le nombre de sorties actives.
- ( c ) L'instruction *with... select* est une instruction concurrente de VHDL.
- ( d ) Si le signal *a* vaut "110", aucune sortie n'est active.
- ( e ) Avec un tel décodeur, il est possible de faire n'importe quel opérateur logique qui comporte au plus 3 entrées et 6 sorties.
- ( f ) Ici, le cas *OTHERS* est obligatoire si l'on veut être certain de fabriquer un circuit combinatoire.
- ( g ) Ici, le cas *OTHERS* est facultatif puisque toutes les combinaisons possibles ont été évoquées.
- ( h ) Il est préférable, mais pas obligatoire, d'utiliser le cas *OTHERS* si l'on veut générer une structure combinatoire.

# QCM

11

Question 10. Parmi les propositions ci-dessous qui se rapportent à la description VHDL suivante, lesquelles sont vraies ?

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY myFunc IS
  PORT (
    a: IN std_logic_vector(7 DOWNTO 0);
    b: IN std_logic_vector(2 DOWNTO 0);
    c: OUT std_logic
  );
END myFunc ;

ARCHITECTURE ar OF myFunc IS
BEGIN
  WITH b SELECT
    c <=
      a(7) WHEN "111",
      a(6) WHEN "110",
      a(5) WHEN "101",
      a(4) WHEN "100",
      a(3) WHEN "011",
      a(2) WHEN "010",
      a(1) WHEN "001",
      a(0) WHEN "000",
      '0' WHEN OTHERS;
END ar;
```

- ( a ) Avec un tel circuit, il est possible de faire n'importe quel opérateur logique qui comporte au plus 3 entrées et une sortie.
- ( b ) Avec un tel circuit, il est possible de faire n'importe quel opérateur logique qui comporte au plus 8 entrées et une sortie.
- ( c ) La description ci-dessus est un multiplexeur 8 vers 1.
- ( d ) La description ci-dessus est un décodeur 1 parmi 8.
- ( e ) La description ci-dessus est un décodeur 3 parmi 8.
- ( f ) La description ci-dessus est un multiplexeur 3 vers 1.
- ( g ) Le cas *OTHERS* n'a aucune chance de se présenter ici, on aurait pu l'enlever.
- ( h ) Même si le cas *OTHERS* n'a aucune chance de se présenter, il ne faut pas le supprimer sinon la description ne correspond plus à un circuit combinatoire.
- ( i ) Le signal *a* correspond à l'adresse de l'entrée sélectionnée pour être en sortie.
- ( j ) Le signal *b* correspond à l'adresse de l'entrée sélectionnée pour être en sortie.

**Question 11** : Quel est le rôle du logiciel Quartus II ?

- ( a ) Transformer une description VHDL en un fichier JEDEC
- ( b ) Transformer un circuit standard en circuit programmable
- ( c ) Transformer un fichier de schéma en description VHDL
- ( d ) Transformer un circuit programmable en circuit programmé
- ( e ) Transformer un fichier C en fichier VHDL

# Corrigés

13

	a	B	C	D	E	F	G	H	i	j	k	l
Q1			X		X							
Q2	x	X		X	X				x		x	
Q3		X					X					
Q4			X									
Q5								x				
Q6					X							
Q7		X	X	X								
Q8			X		X							
Q9	X		X	X		X						
Q10	X		X					x		x		
Q11			X									

# Exercice 0

14

Que fait ce programme ?

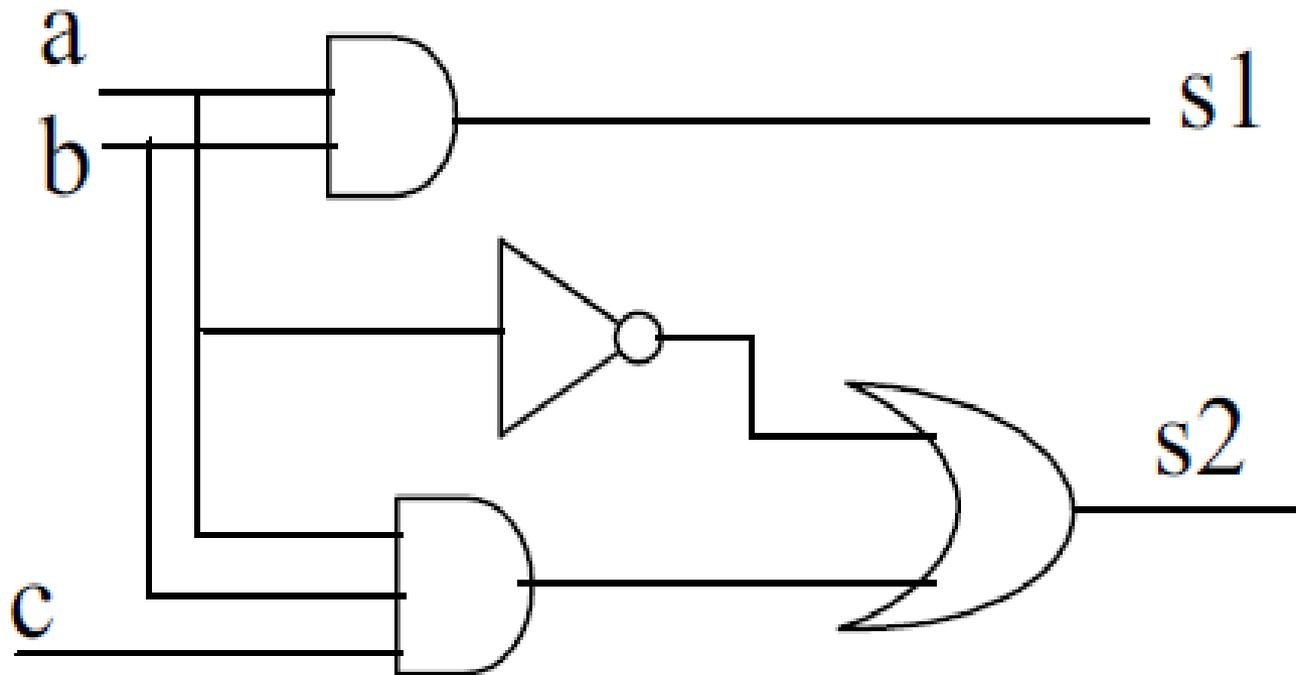
```
LIBRARY ieee;  
USE ieee.std_logic_1164.all;  
  
ENTITY maj IS  
PORT(a,b,c: IN std_logic; m: OUT std_logic);  
END maj;  
--Dataflow style architecture  
ARCHITECTURE concurrent OF maj IS  
BEGIN  
  WITH a & b & c SELECT  
  m <= '1' WHEN "110"|"101"|"011"|"111",  
    '0' WHEN OTHERS;  
END concurrent;
```

Pour exprimer un OU booléen le symbole | est utilisé

# Exercice 1

15

Écrire le programme vhdl qui décrit le schéma ci-dessous



# Réponse Ex1

16

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
ENTITY ex1 IS                                     -- commentaires
PORT (                                           -- déclaration des entrées/sorties physiques
    a, b, c :      IN STD_LOGIC ;
    s1, s2 :      OUT STD_LOGIC  -- pas de point virgule
);
END ex1 ;
ARCHITECTURE numero1 OF ex1 IS  -- description de l'application
BEGIN
    s1 <= a AND b ;
    s2 <= (NOT a) OR (a AND b AND c) ;
END numero1 ;
```

Que se passe-t-il si on  
change l'ordre de S1 et S2 ?

# Réponse Ex1

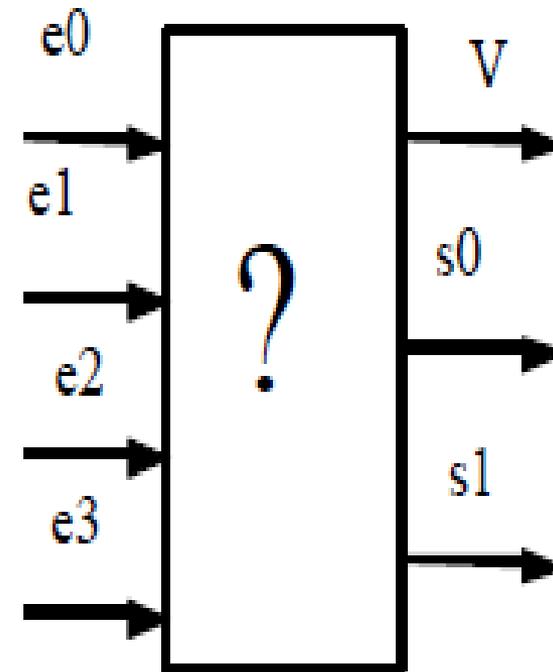
17

- ❑ **Commentaires :**
  - ❑ **Les deux instructions S1 et S2 sont concurrentes :  
L'ordre dans lequel elles sont écrites n'a pas  
d'importance puisqu'au final il s'agit d'associer des  
portes logiques.**
  - ❑ **Contrairement à C ou Pascal, rien ne sera exécuté.**

# Exercice 2

18

- ❑ Décrire l'encodeur de priorités suivant :
  - ❑ Si une seule des entrées est au niveau 1, alors  $V = 1$  et  $S1S0$  indique en binaire le rang  $i$  de cette variable d'entrée.
  - ❑ Si toutes les entrées sont au niveau 0 alors  $V = 0$  et la valeur de  $S1S0$  est 0.



# Réponse Ex2

19

```
LIBRARY ieee;

ENTITY num1 IS
    PORT (
        e :IN STD_LOGIC_VECTOR (3 DOWNTO 0);
        v :OUT STD_LOGIC;
        S :OUT STD_LOGIC_VECTOR (1 DOWNTO 0)
    );
END num1;

ARCHITECTURE archi OF num1 IS
BEGIN

    PROCESS (e)
        BEGIN
            IF e(3) = '1' THEN s = "11"; V= '1';
            ELSIF e(2) = '1' THEN s = "10"; V= '1';
            ELSIF e(1) = '1' THEN s = "01"; V= '1';
            ELSIF e(0) = '1' THEN s = "00"; V= '1';
            ELSE s = "00"; V= '0';
            END IF;
        END PROCESS;
END archi;
```

# Exercice 3

20

- ❑ Décrire le système suivant :
  - ❑ A et B sont des nombres de 8 bits.
  - ❑ Le montage fournit :
    - ❑  $A + B$  (somme), si la commande X vaut 0,
    - ❑  $A - B$ , si cette commande vaut 1.

# Réponse Ex3

21

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
ENTITY num2 IS
    PORT (
        a :IN INTEGER RANGE 0 TO 255;
        b :IN INTEGER RANGE 0 TO 255;
        S :OUT INTEGER RANGE 0 TO 255;
        X :IN STD_LOGIC
    );
END num2;

ARCHITECTURE archi OF num2 IS
BEGIN
    S <= a+b WHEN x = '0' ELSE a-b;
END archi;
```

# Exercice 4

22

Décrire un comparateur 8 bits (fournissant une unique sortie, cette sortie est à 1 si les 2 nombres A et B sont égaux).

# Réponse Ex4

23

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
ENTITY num3 IS
    PORT (
        a :IN INTEGER RANGE 0 TO 255;
        b :IN INTEGER RANGE 0 TO 255;
        S : OUT STD_LOGIC
    );
END num3;

ARCHITECTURE archi OF num3 IS
BEGIN
    S <= '1' WHEN a = b ELSE '0';
END archi;
```

# Exercice 5

24

A est un nombre de 8 bits. Décrire le système qui fournit :

$$A+25, \quad \text{si } A > 9$$

$$A+45, \quad \text{si } A < 10$$

(résultat sur 8 bits).

# Réponse Ex5

25

```
ENTITY num4 IS
    PORT (
        a :IN INTEGER RANGE 0 TO 255;
        S :OUT INTEGER RANGE 0 TO 255
    );
END num4;

ARCHITECTURE archi OF num4 IS
BEGIN
    S <= a + 25 WHEN a > 9 ELSE a + 45;
END archi;
```

# Exercice 6

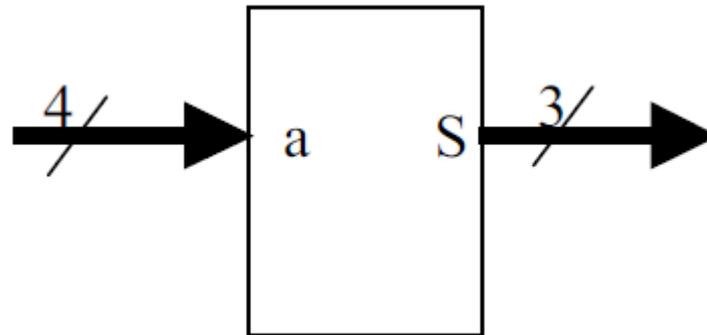
26

Le cahier des charges est le suivant : le nombre  $S$  en sortie indique, en binaire, le nombre de signaux d'entrée à 1.

Si  $a = 1001$  alors  $S = 010$

Si  $a = 1100$  alors  $S = 010$

Si  $a = 1111$  alors  $S = 100$



# Réponse Ex6

27

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
ENTITY combi7 IS
    PORT (
        a: IN STD_LOGIC_VECTOR(3 DOWNTO 0);
        S : OUT INTEGER RANGE 0 TO 4
    );
END combi7;

ARCHITECTURE archi OF combi7 IS
BEGIN
    PROCESS (a)
    BEGIN
        resultat := 0;
        FOR i IN 0 TO 3 LOOP
            IF (a (i) = '1') THEN
                resultat := resultat + 1;
            END IF;
        END LOOP;
    END PROCESS;
END archi;
```

Que manque-t-il à ce programme pour qu'il soit correct ?

# Réponse Ex6

28

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
ENTITY combi7 IS
    PORT (
        a: IN STD_LOGIC_VECTOR(3 DOWNTO 0);
        S : OUT INTEGER RANGE 0 TO 4
    );
END combi7;
```

```
ARCHITECTURE archi OF combi7 IS
BEGIN
    PROCESS (a)
        VARIABLE resultat : INTEGER;
    BEGIN
        resultat := 0;
        FOR i IN 0 TO 3 LOOP
            IF (a (i) = '1') THEN
                resultat := resultat + 1;
            END IF;
        END LOOP;
    END PROCESS;
END archi;
```

```
END LOOP;
s <= resultat;
END PROCESS;
END archi;
```

**Le type de la variable resultat n'a pas été déclaré**

