

1) Quel est l'extension du fichier créée par la synthèse, pour la programmation du FPGA ?

- 1 .bit
- 2 .txt
- 3 .ise
- 4 .xise
- 5 .vhd
- 6 .exe

2) Citer un autre langage permettant de programmer un FPGA (autre que VHDL).

- 1 C/C++
- 2 JavaScript
- 3 WebGL
- 4 Verilog
- 5 Java
- 6 python

3) Trouver la proposition qui correspond au mieux au process VHDL à droite :

- 1 Le process est actif uniquement sur front montant
 - 2 Le process est actif uniquement sur front descendant
 - 3 Il s'agit d'une bascule D avec un multiplexeur
 - 4 Il s'agit d'un compteur avec un démultiplexeur
 - 5 Le signal Rst est synchrone
 - 6 Si le signal sel bouge on peut avoir une modification de S en fonction de Sel
- ```

Process(Rst, clk)
begin
If (Rst = '1') then
S <= "0000";
elsif (clk='1' and clk'event) then
If (sel = '1') then
S <= A;
else
S <= B;
End if;
End if;
End process;

```

4) Le constructeur Xilinx propose actuellement d'autres outils (autre que ISE) pour le développement sur FPGA : lequel ou lesquels

- 1 Vivado
- 2 HDL Coder
- 3 freeScale
- 4 Quartus
- 5 Leonardo
- 6 Cadence

5) Quel est l'opérateur qui réalise la concaténation ?

- 1 \$
- 2 &&
- 3 %
- 4 \*\*
- 5 &
- 6 <=

6) Un composant en VHDL synthétisable xdxest au minimum composé :

- 1 d'une architecture
- 2 d'une instance de composant
- 3 d'un process
- 4 d'instructions concurrentes
- 5 d'une entité
- 6 d'une horloge

7) Examiner le code VHDL à droite et cocher les bonnes réponses :

- 1 il y a une erreur sur : "when others" il manque quelque chose, par exemple S <= "ZZZZ";
  - 2 il y a juste un "warning" sans erreur, car il y a deux signaux asynchrones dans ce process
  - 3 il y a une erreur ici : "when "10" | "11" =>" cela n'existe pas cette barre
  - 4 il manque un ";" ou il y en a un en trop quelque part
  - 5 A, doit être un std\_logic\_vector de 4 bits
  - 6 & : il s'agit d'un « et » numérique
- ```

Rst_En_signal <= Rst & En;
Process(Rst_En_signal, clk)
begin
case Rst_En_signal is
when "10" | "11" =>
S <= "0000";
when "01" =>
S <= "1111";
when "00" =>
if (clk='1' and clk'event)
then
S <= A;
end if;
when others =>
end case;
End process;
    
```

8) La description d'une entité peut contenir

- 1 La déclaration du port
- 2 La déclaration d'un generic
- 3 La déclaration des signaux internes
- 4 La déclaration d'un process
- 5 La déclaration des composants utilisés
- 6 La déclaration de library

9) En VHDL entre Architecture et Begin, on peut trouver :

- 1 Des déclarations de composants
- 2 Des déclarations de signaux internes
- 3 Des instances de composants
- 4 Des instructions concurrentes
- 5 Des instructions séquentielles
- 6 La déclaration d'une entité

10) En VHDL deux process différents peuvent modifier le même signal

- 1 Non
- 2 Oui

11) En VHDL peut-il avoir un process dans un process ?

- 1 Non
- 2 Oui

12) Examiner le composant VHDL à droite :

- 1 Ce code est synthétisable, il n'y a pas d'erreur, il s'agit d'une bascule D active sur niveau haut de B
 - 2 Ce code n'est pas synthétisable, il y a une erreur
 - 3 Le composant créé est combinatoire car il n'y a pas de process
 - 4 Le composant créé est séquentiel asynchrone
 - 5 Le composant créé est analogique
 - 6 Le composant créé est séquentiel synchrone
- ```

Library IEEE;
Use IEEE.Std_Logic_1164.all;
Entity TOTO is
port(A,B : in std_logic;
Q : out std_logic);
end TOTO;
Architecture TITI of TOTO is
signal Q_signal : std_logic;
Begin
Q_signal <= A when B='1' else Q_signal;
Q <= Q_signal;
end;

```

18) En VHDL, les instructions séquentielles sont :

- 1 Une affectation permanente
- 2 Une affectation permanente avec when/else ou with/select
- 3 Une affectation différée
- 4 Une affectation différée avec if/then/else, case/when

19) Les instructions Wait et After sont-elles synthétisables ?

- 1 Non
- 2 Oui

20) La signification de FPGA

- 1 Flat Package Gate Addressable
- 2 Field Programmable Gate Arrays
- 3 Fabrication Processor Gate American
- 4 First Professional Gate Arrays

13) Examiner le composant VHDL à droite :

```
Library IEEE;
Use IEEE.Std_Logic_1164.all;
Entity TOTO is
Port(A,B : in std_logic;
Q : out std_logic);
end TOTO;
Architecture TITI of TOTO is
Begin
Q <= A when B='1' else not A;
end;
```

- 1 Ce code est synthétisable, il n'y a pas d'erreur, il s'agit d'une porte non xor
- 2 Ce code n'est pas synthétisable, il y a une erreur
- 3 Le composant créé est 100% combinatoire
- 4 Le composant créé est analogique
- 5 Le composant créé est séquentiel synchrone
- 6 Il s'agit d'une bascule D

14) Examiner le process VHDL à droite :

```
process(A,B)
begin
if (A='1') then
Q<=B;
end if;
end process;
```

- 1 Ce code est synthétisable, il n'y a pas d'erreur, il y a une bascule D active sur niveau haut de A
- 2 Ce code n'est pas synthétisable, il y a une erreur
- 3 Le composant créé est 100% combinatoire
- 4 Le composant créé est séquentiel
- 5 Le composant créé est analogique
- 6 Il s'agit d'une bascule D sur front

15) Examiner le process VHDL à droite :

```
process(A,B)
begin
if (A='1') then
Q<=B;
else
Q<=0';
end if;
end process;
```

- 1 Ce code est synthétisable, il n'y a pas d'erreur, il s'agit d'une porte AND
- 2 Ce code n'est pas synthétisable, il y a une erreur
- 3 Le composant créé est 100% combinatoire
- 4 Le composant créé est séquentiel et asynchrone
- 5 Le composant créé est analogique
- 6 Le composant créé est séquentiel synchrone

16) En VHDL, dans un process, on peut trouver

- 1 Une architecture
- 2 Des instructions séquentielles
- 3 Un autre process
- 4 Des instructions concurrentes
- 5 Des instances de composants
- 6 un port

17) En VHDL, les instructions concurrentes sont :

- 1 Une affectation permanente
- 2 Une affectation permanente avec when/else ou with/select
- 3 Une affectation différée
- 4 Une affectation différée avec if/then/else, case/when
- 5 Exécutées l'une après l'autre
- 6 Obligatoirement combinatoire

**FORMULAIRE RÉPONSE**

**Bien noircir les cases justes  
en cas d'erreur blanchir sans refaire le cadre**

|          |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |
|----------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| Nom :    |                          | note :                   |                          |                          |                          |                          |                          |                          |                          |                          |
| Prénom : |                          |                          |                          |                          |                          |                          |                          |                          |                          |                          |
|          | 1                        | 2                        | 3                        | 4                        | 5                        | 6                        |                          |                          |                          |                          |
| Q1       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Q2       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Q3       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Q4       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Q5       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Q6       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Q7       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Q8       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Q9       | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Q10      | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |