



Sciences et technologies de l'Industrie et du développement durable

PARCOURS DE FORMATION SIN – FPGA

Formation des enseignants.
Spécialité SIN du baccalauréat STI2D

Christian Dupaty

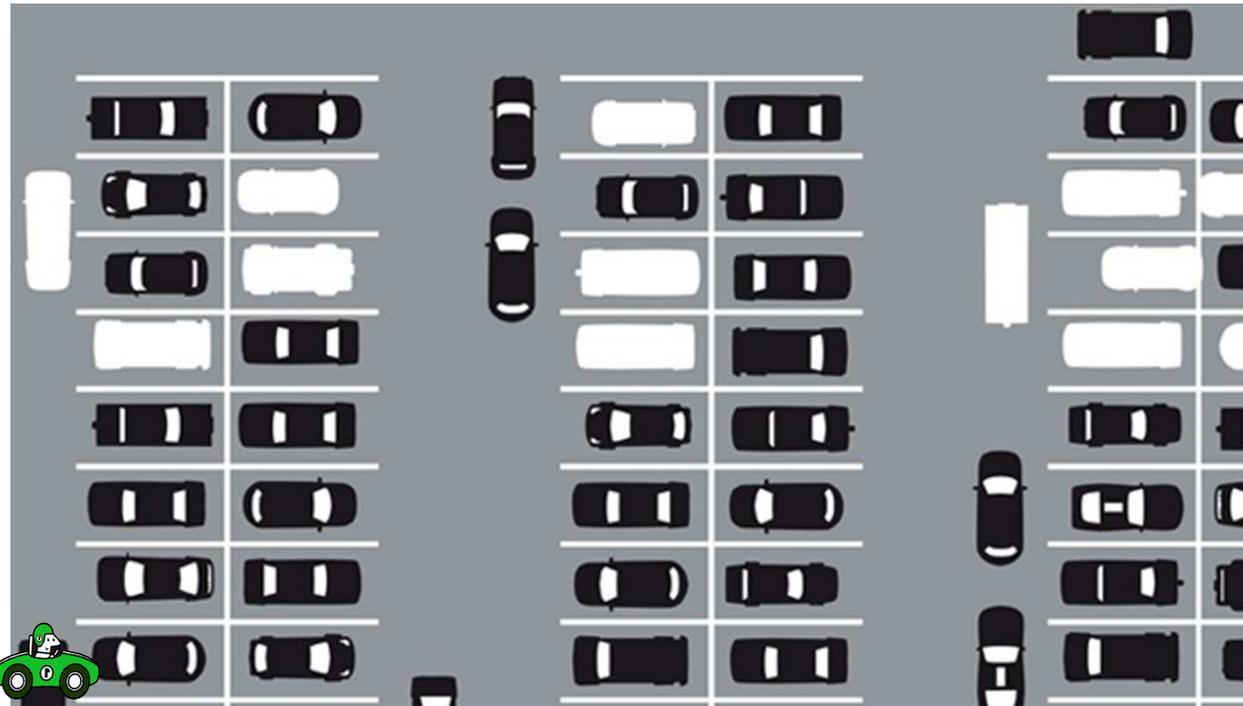
Professeur de génie électrique
Lycée Fourcade 13120 Gardanne

christian.dupaty@ac-aix-marseille.fr

<http://genelaix.free.fr>

Contrôle D'ACCES à un parking

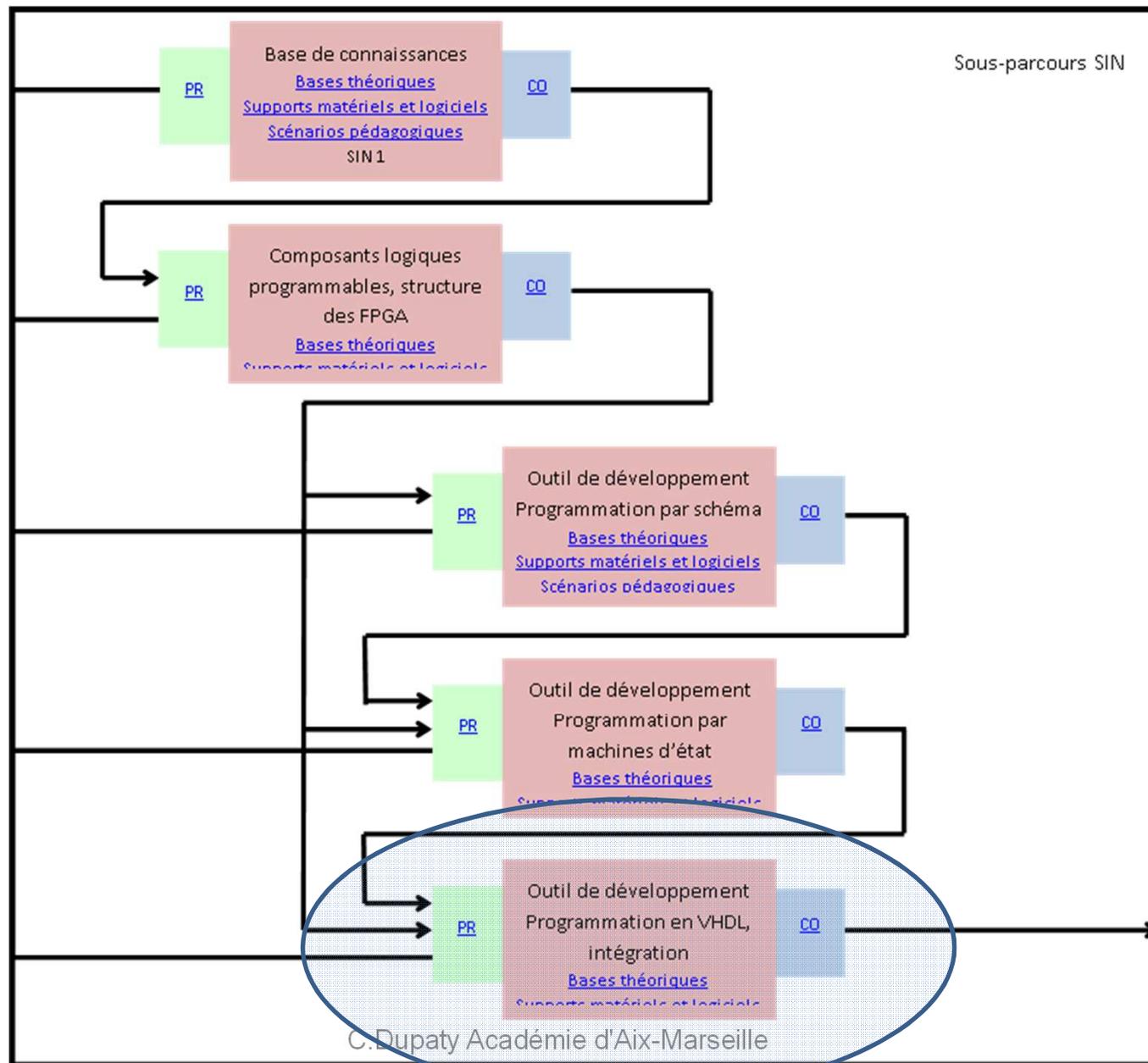
Descriptions en VHDL



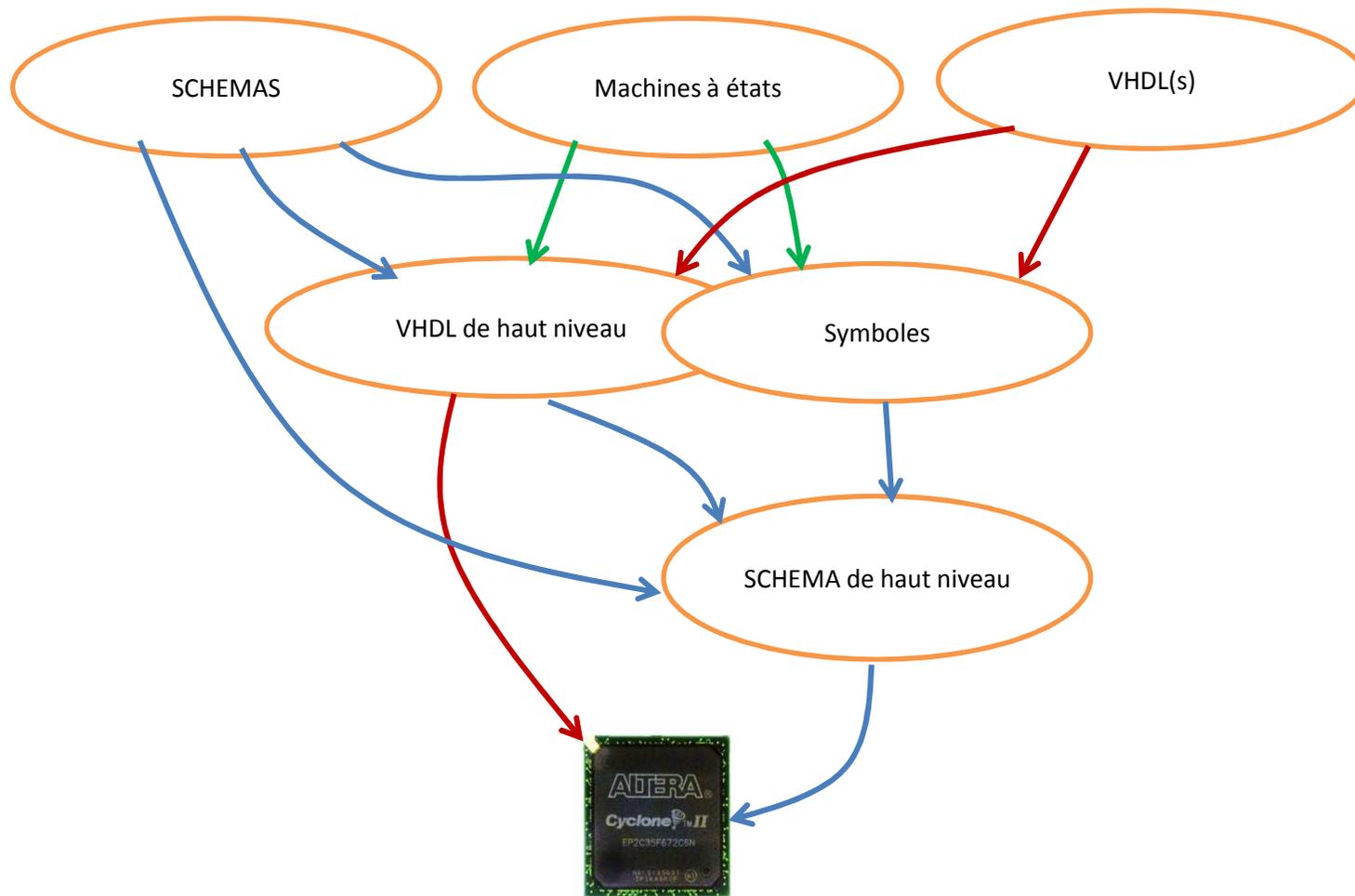
Projet support du parcours SIN FPGA

Christian Dupaty – académie d'Aix-Marseille
christian.dupaty@ac-aix-marseille.fr <http://genelaix.free.fr/>

Parcours SIN-FPGA



Développement d'applications sur FPGA



VHDL

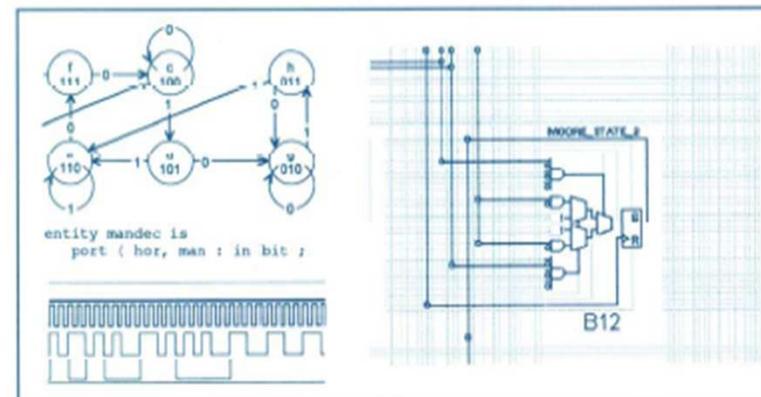
- Les TP font appels à des notions du langage VHDL.
- Consultez le livre de J.Weber et S.Moutault
<http://books.google.fr/books?id=AKoI0wjcnUC>

COLLECTION TECHNOLOGIES

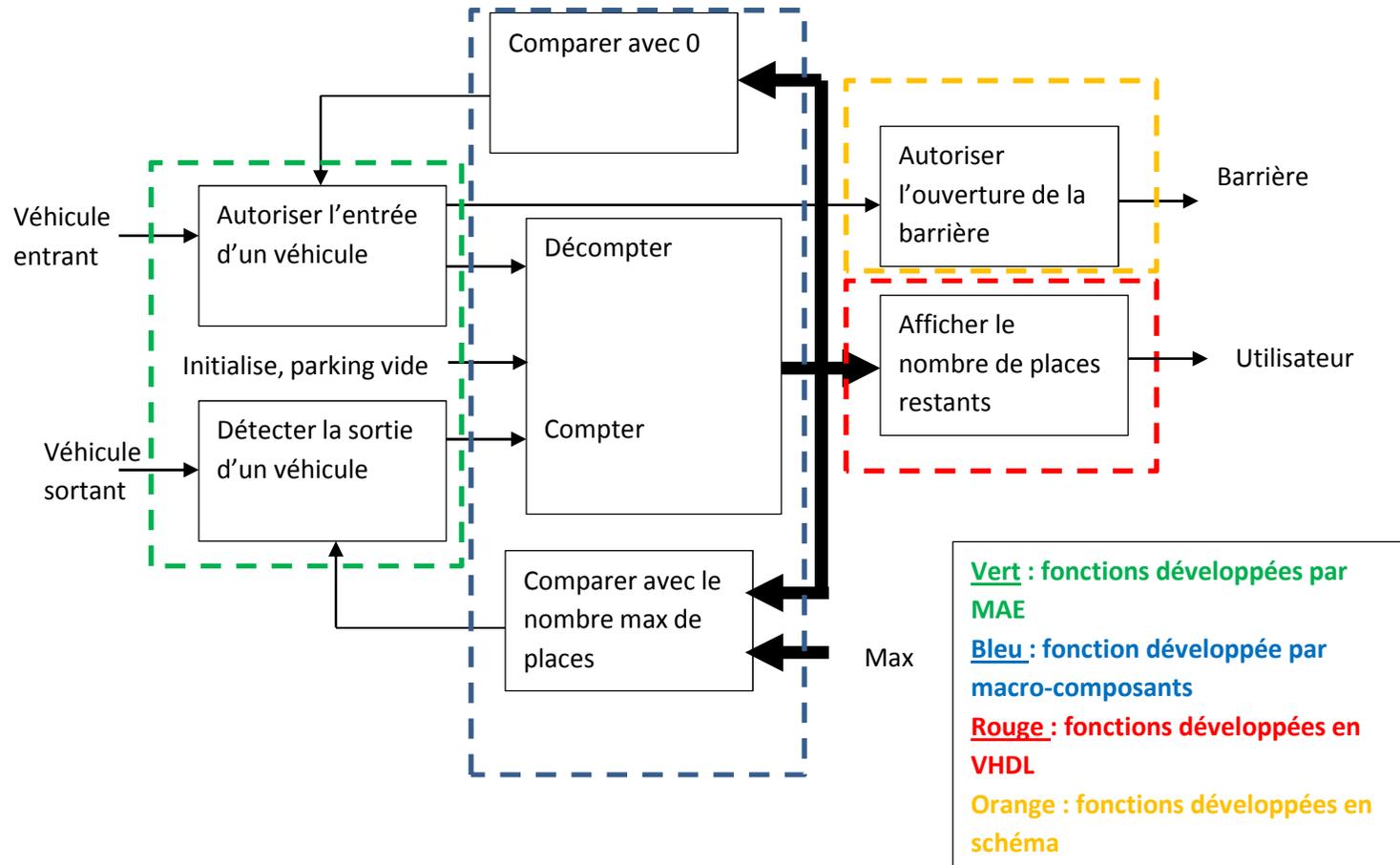
circuits numériques et synthèse logique un outil: VHDL

J. WEBER

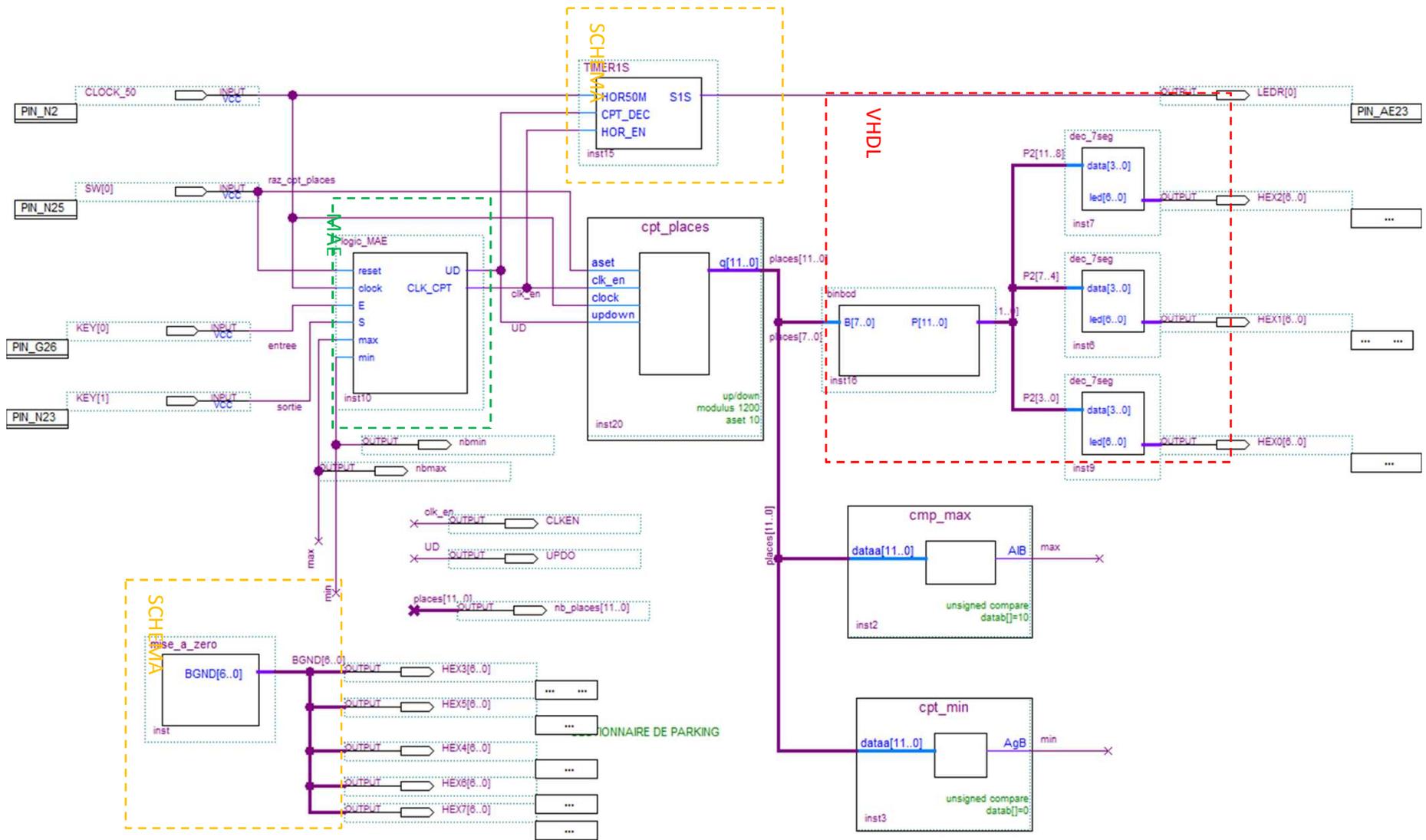
M. MEAUDRE



Parking - Descriptions



Structure logique à réaliser



VHDL: décodeur 2 → 4

Description par équations

E1	E0	S3	S2	S1	S0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

-- VHDL Decodeur 2 vers 4
 -- Description par equations

Library ieee;
 Use ieee.std_logic_1164.all;

} Déclaration des bibliothèques de fonctions utilisées

ENTITY dec24 is
 PORT (

S
 E
 END dec24;

: OUT STD_LOGIC_VECTOR (3 downto 0);
 : IN STD_LOGIC_VECTOR (1 downto 0);

Le nom des signaux qui rentrent et sortent du composant

Ici deux vecteurs de bit (en fait des bus) un de deux et un de trois bits

E contient les bits E(0) et E(1)

ARCHITECTURE combinatoire OF dec24 IS
 signal S0, S1, S2, S3 : STD_LOGIC ;
 BEGIN

S0 <= not(E(0)) and not (E(1)) ;
 S1 <= E(0) and not(E(1));
 S2 <= not(E(0)) and E(1);
 S3 <= E(0) and E(1) ;
 S <= S3 & S2 & S1 & S0;

Équations logiques :

Dans ce mode de description c'est le programmeur qui résout le problème. On retrouve ici les fonctions logiques d'un schéma.

END combinatoire;

VHDL: décodeur 2 → 4

Description when - else

-- Decodeur 2-4 avec when else

Library ieee;

Use ieee.std_logic_1164.all;

ENTITY Decode is

PORT (

S : OUT STD_LOGIC_VECTOR (3 downto 0);

E : IN STD_LOGIC_VECTOR (1 downto 0));

END Decode;

ARCHITECTURE combinatoire OF Decode IS

BEGIN

S <= "0001" WHEN E = "00"

else "0010" WHEN E = "01"

else "0100" WHEN E = "10"

else "1000" ;

END combinatoire;

VHDL: décodeur 2 → 4

Description with - select

```
-- Decodeur 2-4 avec with select  
Library ieee;  
Use ieee.std_logic_1164.all;
```

```
ENTITY Decode is  
    PORT (  
        S      : OUT STD_LOGIC_VECTOR (3 downto 0);  
        E      : IN  STD_LOGIC_VECTOR (1 downto 0));  
END Decode;
```

```
ARCHITECTURE combinatoire OF Decode IS  
    signal S0, S1, S2, S3 : STD_LOGIC ;  
begin  
    with E select  
        S <=    "0001" when "00",  
               "0010" when "01",  
               "0100" when "10",  
               "1000" when "11",  
               "----" when others ;  
END combinatoire;
```

VHDL: décodeur 2 → 4

Description par process (if , case ..)

```
-- Decodeur 2-4 avec process
```

```
Library ieee;
```

```
Use ieee.std_logic_1164.all;
```

```
ENTITY Dec2_4_process is
```

```
    PORT (
```

```
        S      : OUT STD_LOGIC_VECTOR (3 downto 0);
```

```
        E      : IN  STD_LOGIC_VECTOR (1 downto 0));
```

```
    END Dec2_4_process;
```

```
ARCHITECTURE combinatoire OF Dec2_4_process IS
```

```
Begin
```

```
    process(E)
```

```
    begin
```

```
        case E is
```

```
            when "00" => S <= "0001";
```

```
            when "01" => S <= "0010";
```

```
            when "10" => S <= "0100";
```

```
            when "11" => S <= "1000";
```

```
            when others => S <= "0000";
```

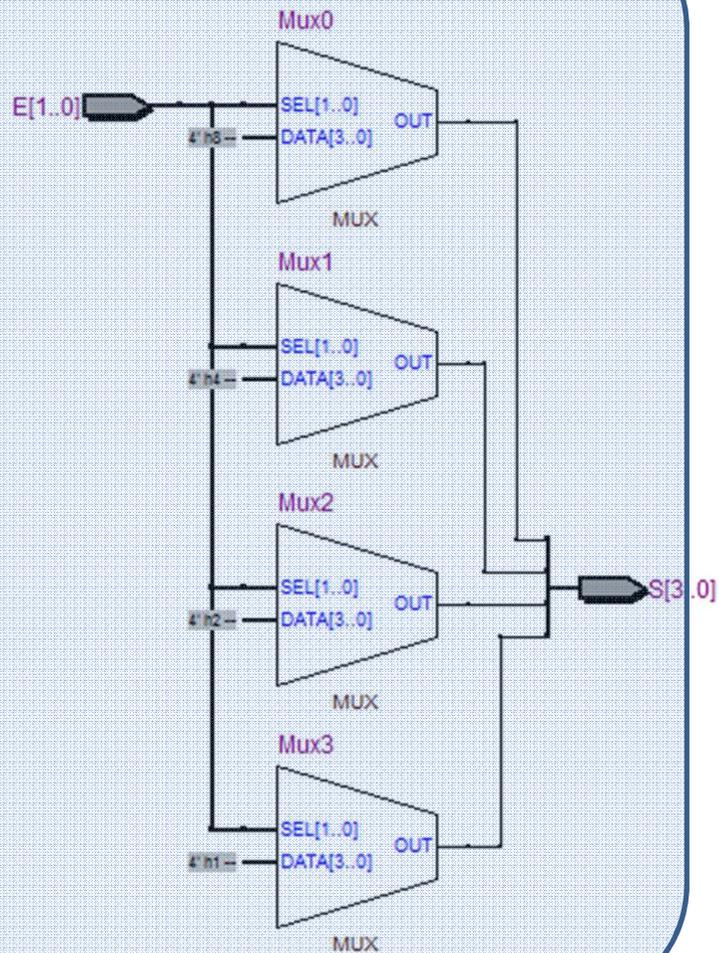
```
        end case;
```

```
    end process;
```

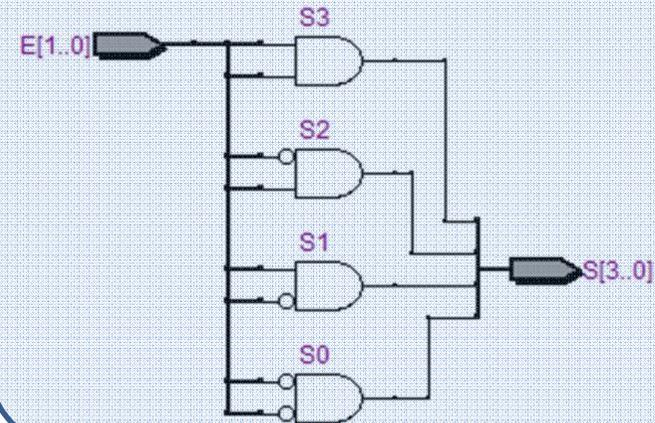
```
end combinatoire;
```

Synthèse VHDL comportementale

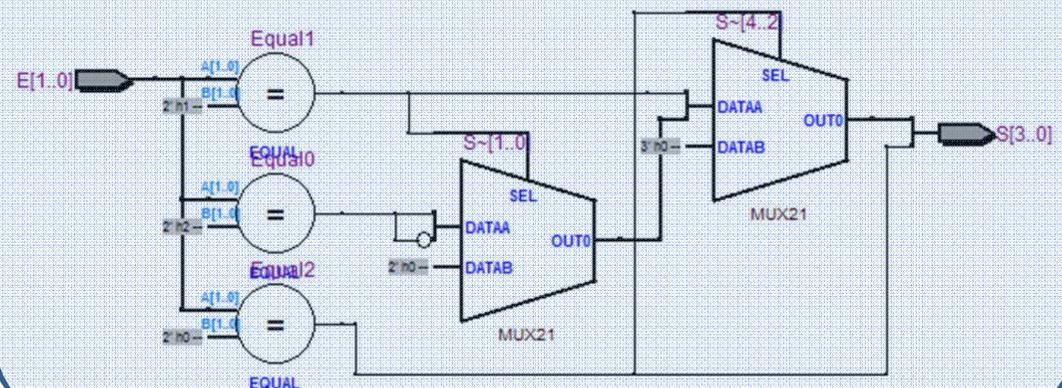
Avec with - select



Par équations

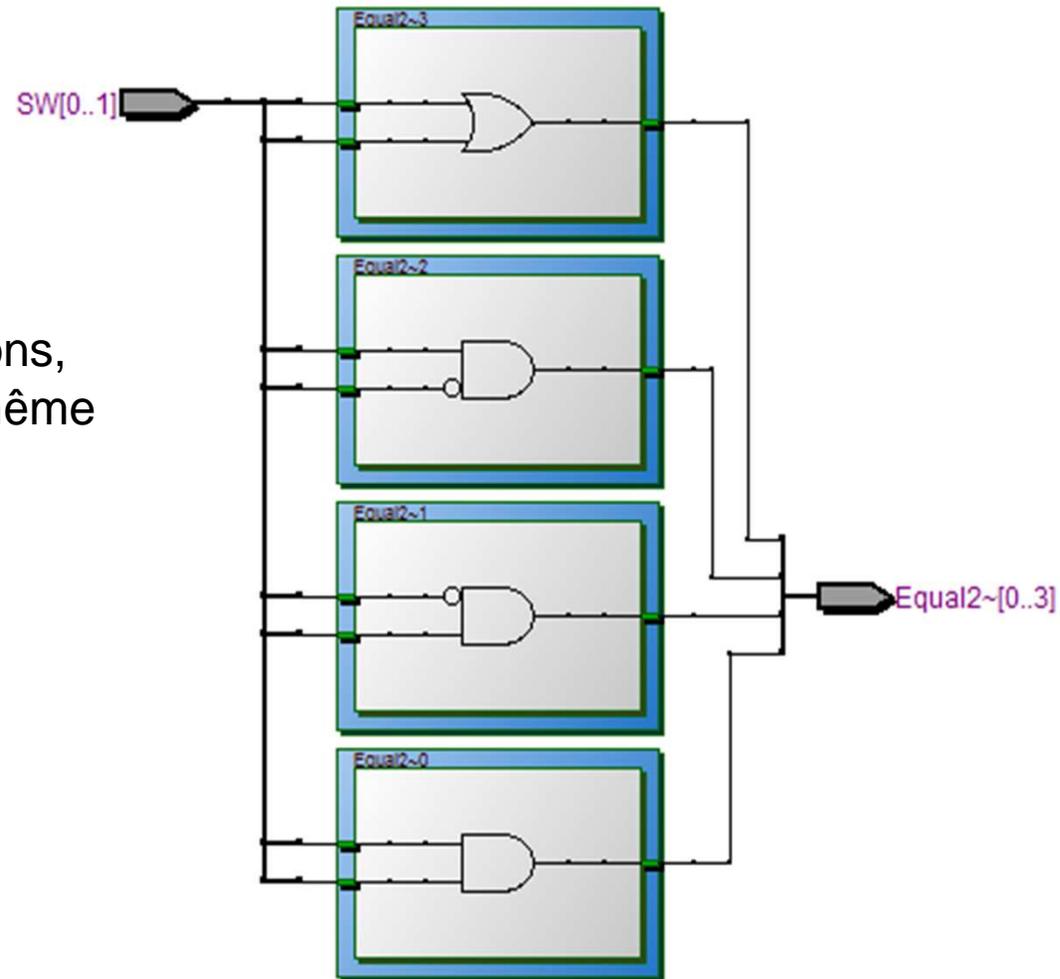


Avec when - else



Synthèse VHDL gates

Dans les quatre descriptions, QUARTUS arrivera à la même optimisation



Exercices

- **Description combinatoire :**
Decodeur 7 segments sur KIT DE2 (projet parking)
- **Description séquentielle :**
clignoteur 1s
- **Analyse d'un programme complexe :**
Horloge sur afficheur LCD, ajout de boutons de réglage des heures et des minutes

Maintenant, il faut essayer...

