



# Utilisation de Quartus II

## sur des exemples simples

---

***Outil logiciel : Quartus II Altéra***

***Outil matériel : carte DE1 Altéra***

# Utilisation de Quartus II

## sur des exemples simples

---

***Premier projet sous Quartus***   ***Porte ET***

***saisie graphique***  
***simulation fonctionnelle***  
***simulation temporelle***  
***programmation dans le FPGA***  
***test***

***Additionneur***

***2x1***  
***2x3***  
***et 2x5***

***Mini Projet***

***Diviseur***  
***Compteur***  
***Transcodeur BCD/7 segments***

***Circuit de génération de PWM***



# Utilisation de Quartus II sur des exemples simples

M2 SME

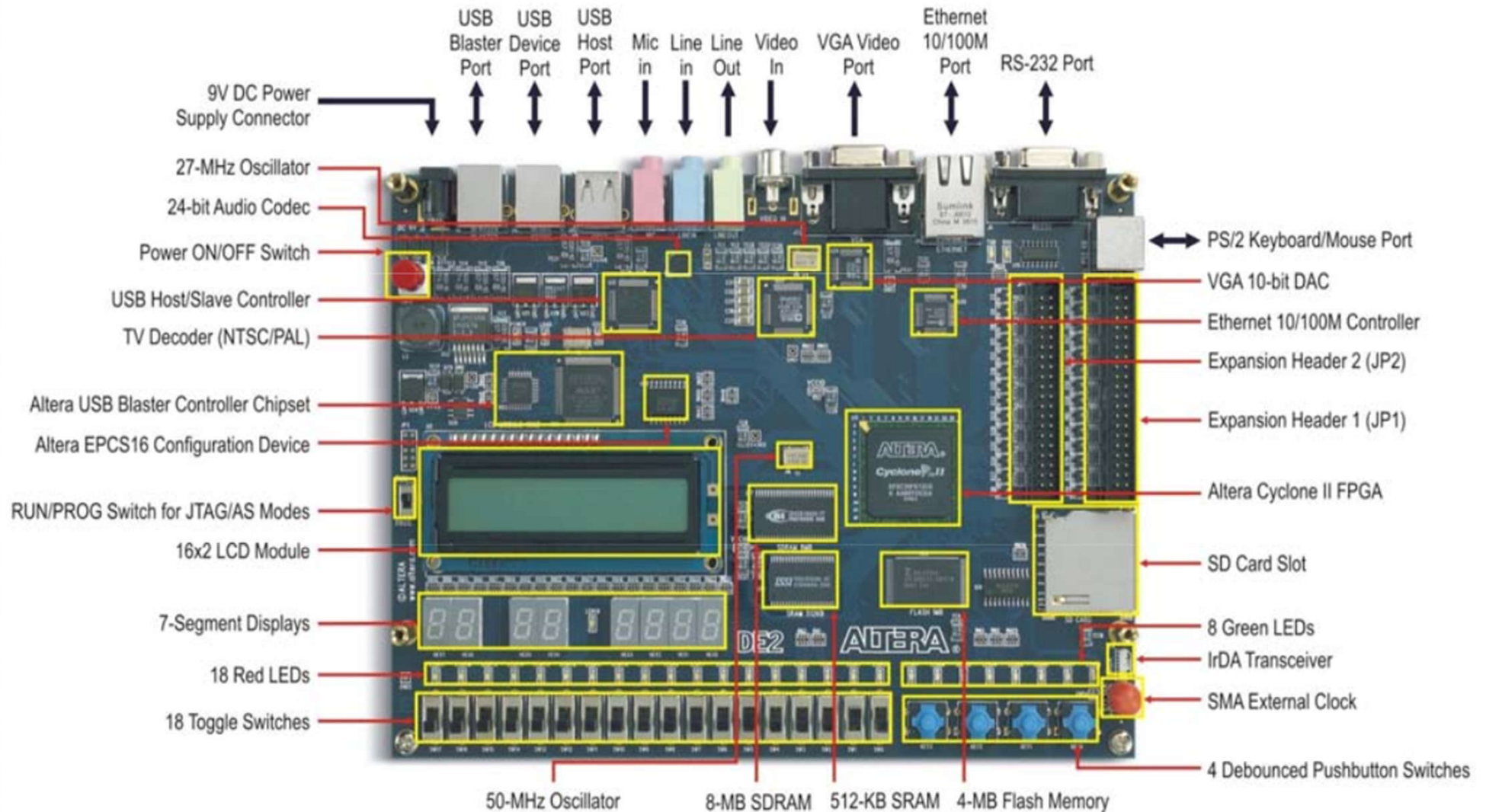
TPs Quartus II

TP



TPs de base Quartus II

# Utilisation de Quartus II sur des exemples simples





# Utilisation de Quartus II

## sur des exemples simples :

---

### ***Premier projet sous Quartus II***

***Réaliser une Porte ET (entrée graphique)***

*en suivant les indications et en se servant du manuel de Quartus II disponible*

***Création d'un nouveau projet (sur Quartus II)***

***saisie graphique de la porte ET***

***Compilation***

*Synthèse logique*

*Synthèse physique*

***Simulations :***

*simulation fonctionnelle*

*simulation temporelle*

***programmation dans le FPGA***

***test sur carte DE2***

# Utilisation de Quartus II

## sur des exemples simples :

**Réaliser un additionneur de deux mots de 1 bit :**

**avec comme entrées : 2 bits A1 et B1 et une retenue d'entrée Cin.  
et comme sorties : 1 bit S1 et une retenue de sortie Cout.**

**Donner la table de vérité de l'additionneur**

**Donner les tableaux de Karnaugh de S1 et de Cout**

**Donner les équations de S1 et de Cout**

**Donner le schéma structurel à l'aide de portes**

**Créer un nouveau projet sur Quartus II appelé TPADD1**

**Créer une fenêtre graphique appelée TPADD1.bdf qui sera le niveau hiérarchique le plus haut de votre projet.**

**Saisir le schéma sur une nouvelle fenêtre graphique appelée ADD1.bdf et créer le symbole de ce fichier.**

**Se mettre sur TPADD1.bdf et rapatrier le symbole de ADD1 et créer les entrées sorties nécessaires.**

**Réaliser la compilation et simulation fonctionnelle de TPADD1 et si OK visualiser les synthèses logique et physique générées par Quartus, enfin n'oublier pas de créer le symbole de TPADD1.**

**Réaliser l'implémentation sur le composant et valider sur la carte DE2, on choisira pour A1, B1 et Cin (SW0, SW1 et SW2) et pour S1 et Cout (LEDR0 et LEDR1).**



# Utilisation de Quartus II

## sur des exemples simples :

***Créer un nouveau projet sur Quartus II appelé TPADD2x3  
Créer une fenêtre graphique appelée TPADD2x3.bdf qui sera le niveau hiérarchique le plus haut de votre projet.***

***Sur une nouvelle feuille graphique TPADD2x3-1.bdf réaliser un additionneur de deux mots de 3 bit en utilisant le symbole de TPADD1:  
avec comme entrées : 2 mots de 3 bits A2 A1 A0 et B2 B1 B0 et une retenue d'entrée Cin.  
et comme sorties : 1 mot de 3 bits S2 S1 S0 et une retenue de sortie Cout.***

***Réaliser la compilation et simulation fonctionnelle de TPADD2x3 et si OK visualiser les synthèses logique et physique générées par Quartus.  
Réaliser l'implémentation sur le composant et valider sur la carte DE1, on choisira pour A0, A1, A2, B0, B1, B2 et Cin (SW0, SW1, SW2, SW3, SW4, SW5 et SW6) et pour S0, S1, S2 et Cout (LEDR0, LEDR1, LEDR2 et LEDR3).***

***Créer un nouveau projet sur Quartus II appelé TPADD2x5  
Créer une fenêtre graphique appelée TPADD2x5.bdf qui sera le niveau hiérarchique le plus haut de votre projet.  
Écrire une description VHDL ADD2x5.vhd réalisant un additionneur (5 bits). Pour cela on impose l'utilisation de vecteurs.  
Programmer et valider sur la carte DE2.***

# Utilisation de Quartus II

## sur des exemples simples

### ***Premier Projet sous Quartus II : Comptage et affichage des secondes de 0 à 9***

#### ***Cahier des charges :***

*On veut réaliser une fonction permettant, à partir d'un oscillateur à quartz de 50Mhz présent sur la carte DE2 de visualiser l'écoulement des secondes sur un afficheur 7 segments.*

*On privilégiera pour ce projet l' **APPROCHE FONCTIONNELLE***

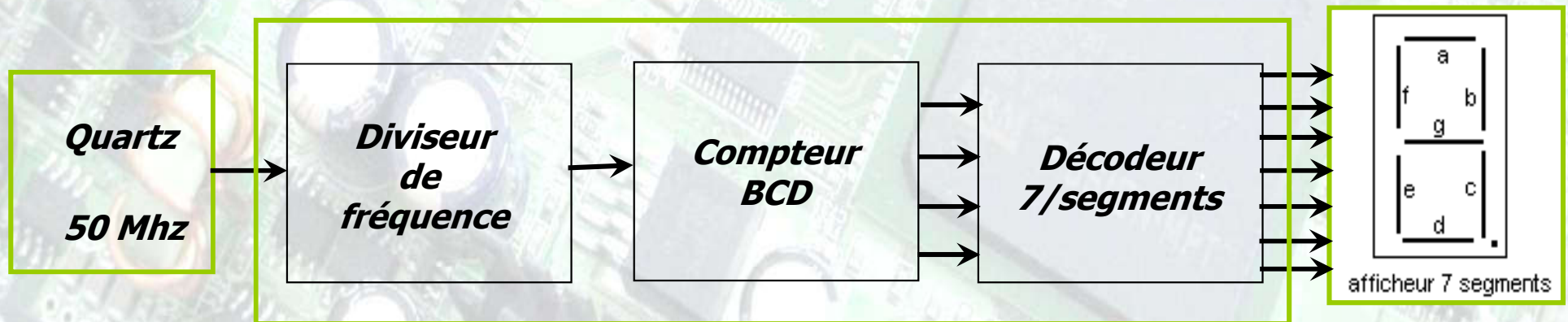




# Utilisation de Quartus II

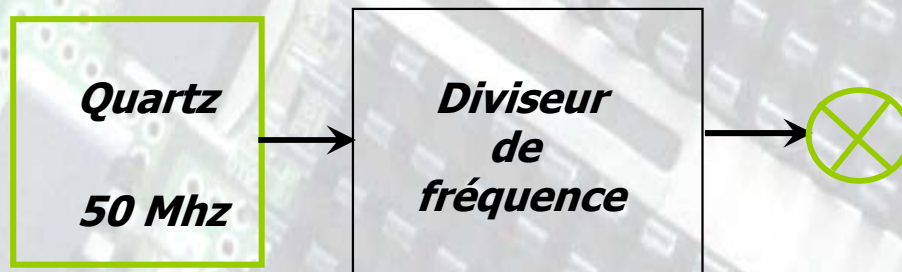
## sur des exemples simples

**Projet :** On veut réaliser une fonction permettant, à partir d'un oscillateur à quartz de 50Mhz présent sur la carte DE2 de visualiser l'écoulement des secondes sur un afficheur 7 segments.



### Génération du signal de 1 Hz :

Dans une première étape on veut réaliser un signal de fréquence 1 Hz à partir de l'horloge de 50 Mhz. Pour cela il faut réaliser un diviseur de fréquence, la sortie de celui-ci sera une led présente sur la carte.



**Réalisez en VHDL le programme du diviseur.**  
**Créer un symbole du diviseur.**

# Utilisation de Quartus II

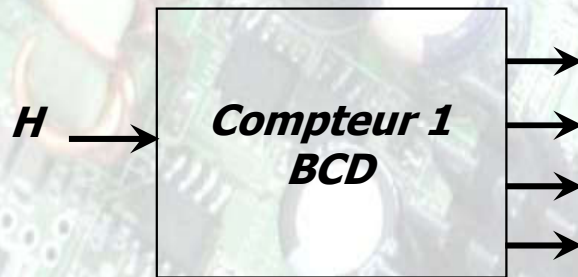
## sur des exemples simples

### Réalisation du compteur BCD :

Dans un premier temps, il est demandé de réaliser un simple compteur BCD, possédant 1 entrée d'horloge H et 4 sorties.

→ (0,1,2,...,15,0,1,..)

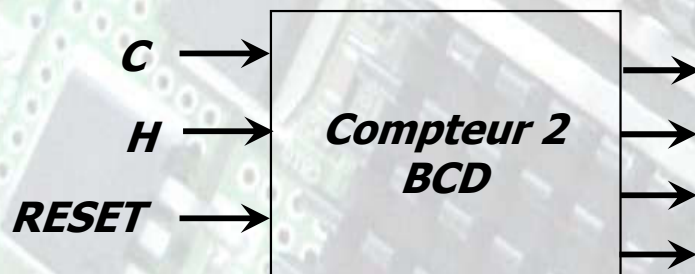
Sur l'entrée H un signal d'horloge et sur les sorties 4 LEDS.



On veut réaliser maintenant un compt/décompt BCD avec un RESET asynchrone

C=1 comptage

C=0 décomptage



**Proposer un schéma structurel du compteur 1 :**  
donner la table de vérité, karnaugh, la réalisation à l'aide de bascules D:7474 et de portes logiques.

**Donner le programme VHDL du compteur 1.**

**a- à partir des nouvelles équations.**

**b- à partir du diagramme d'état.**

**c- à partir de l'analyse comportementale.**

Compilation et simulation fonctionnelle.

Intégration et validation sur le composant.

**Donner le programme VHDL du compteur 2 à partir de l'analyse comportementale.**

Compilation et simulation fonctionnelle.

Intégration et validation sur le composant.



# Utilisation de Quartus II

## sur des exemples simples

### Réalisation du compteur BCD : (Amélioration)

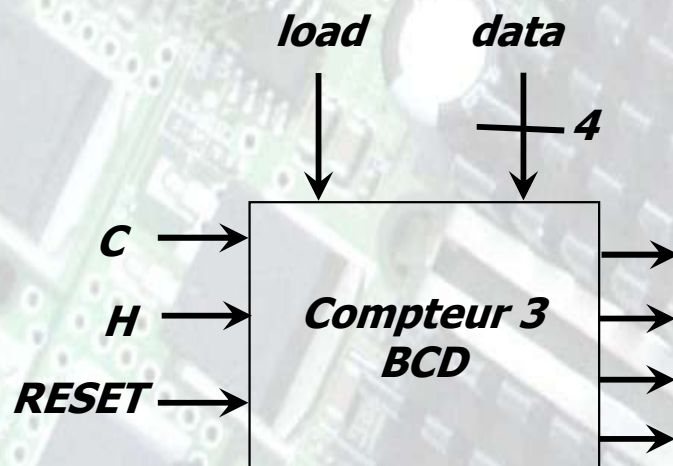
*Cahier des charges :*

*On veut réaliser un compteur/décompteur*

*modulo 10 (0,1,2,...,9,0,1,...) pour C=1*

*RESET asynchrone*

*Pré-chargement synchrone (load)*



**Donner le programme VHDL du compteur 3 à partir de l'analyse comportementale.**

*Compilation et simulation fonctionnelle.  
Intégration et validation sur le composant.*

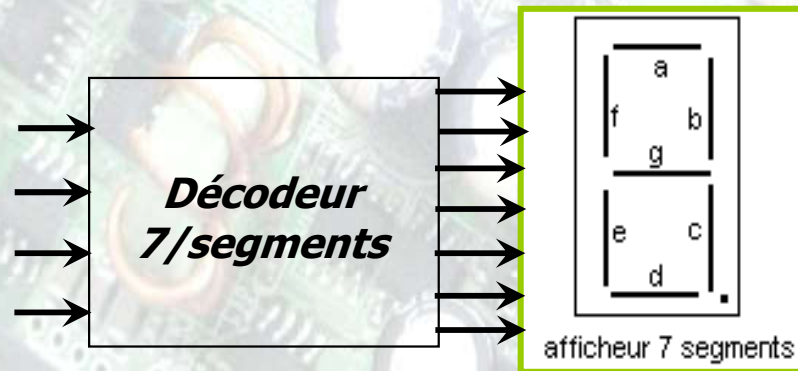
**Créer un symbole du compteur.**

# Utilisation de Quartus II

## sur des exemples simples

### Réalisation du décodeur BCD / 7segments :

4 entrées : 4 interrupteurs présents sur la carte  
7 sorties : afficheur 7 segments.



**Réaliser un décodage de 0 à 9, pour toutes les autres combinaisons présentes à l'entrée nous afficherons un E signalant une erreur.**

**Proposer un schéma structurel d'un décodeur BCD/7segments :**  
donner la table de vérité, le tableau de karnaugh, les équations et le schéma structurel de ce décodeur.

**Donner les différents programmes VHDL de ce décodeur en utilisant :**

- des instructions du mode concurrent :

- Affectation inconditionnelle
- Affectation conditionnelle
- Affectation sélective

- des instructions du mode séquentiel :

- Assignment conditionnelle
- Assignment sélective

**segA <= équation(A,B,C,D)**

**when else**

**with select when**

**if then (elsif then ) (else ) end if;**  
**case is when when others end case;**

Compilation et simulation fonctionnelle de chacun des programmes.

Intégration et validation sur le composant.



# Utilisation de Quartus II

## sur des exemples simples

---

### **Génération d'une PWM :**

#### **Cahier des charges d'un circuit de génération de PWM**

Un circuit générateur PWM comprend :

- Un compteur libre sur N bits piloté par une horloge de référence **clk**.
- Un comparateur sur N bits qui compare la sortie du compteur avec son modulo (**FREQ** : ce qui permet de fixer la fréquence de la PWM). La sortie du comparateur assure la remise à zéro du compteur.
- Un comparateur sur N bits qui compare la sortie du compteur avec le rapport cyclique désiré (**DUTY**). La sortie de ce comparateur génère la sortie **pwm\_out**.
- Une entrée **reset\_n** RAZ asynchrone active à 0.

Pour le BE N=16 bits

Donner une description fonctionnelle de la PWM (schéma fonctionnel)

Donner un code VHDL de cette description (avec 2 process)

Réaliser le projet sur Quartus II, le compiler, le simuler et tester sur la carte DE2.