

*Il y a 4 TP de 4 heures pour le Parcours Systèmes autonomes
Il y a 5 TP de 4 heures pour le Parcours Systèmes et logiciels embarqués
Tous les TP sont évalués en séance*

TP 1 : Prise en main de Quartus

Conception de circuits en schématique

Introduction

L'objectif de ce premier TP est la prise en main d'un outil de modélisation, de conception, de description, de simulation et d'implémentation de circuits numériques dans des FPGA (*field-programmable gate array*, réseau de portes programmables), à savoir : **Quartus**.

Cet outil est téléchargeable librement dans sa version web à partir du lien ci-dessous

[Quartus 19.1](#)

Nous utilisons la version **Quartus 15.0** avec une souscription.

Pour ces téléchargements, il fallait ajouter **SoC Embedded Design Suite (EDS)** retrouvé dans Additional Software. Ce qui est proposé est exactement ce qui est présent sur le site de l'ENSEM.

Les TP comportent des parties pratiques où il est demandé d'implémenter des circuits numériques dans un FPGA. La carte DE1-SoC utilisée est une carte d'expérimentation, équipée d'un FPGA du type Cyclone V.

**DE1-SoC Cyclone V SoC 5CSEMA5F31C6
85K Programmable logic elements (LEs)**

Un certain nombre de documents sont à votre disposition pour l'ensemble des TP.

User manuel de la carte DE1-SoC : *DE1-SoC_UserManual.pdf*

Fichier d'assignation globale : *DE1_SoC_Default.qsf*

Fichier de prise en main du logiciel : *Quartus_II_Introduction.pdf*

Dans la première partie du TP, il vous est proposé de suivre pas à pas les étapes présentées dans le tutorial intitulé '*Quartus_II_Introduction.pdf*'.

Dans la seconde partie de ce TP il vous sera demandé de réaliser en schématique un demi-additionneur, un additionneur 1 bit puis un additionneur 4 bits.

Les notions abordées dans ce premier TP sont la création d'un projet, la simulation fonctionnelle avec Qsim (outil de simulation intégré à Quartus) ainsi que l'implémentation d'un circuit numérique dans un FPGA.

Pour les TP du module, vous créez un répertoire général « conception » qui contiendra pour ce TP en particulier un répertoire « TP1 ». Au sein de ce répertoire TP1, vous pouvez créer deux autres pour chacune des 2 parties du TP. Garder tous les fichiers, vous pourrez les utiliser d'un TP à l'autre.

Partie I : Prise en main de Quartus Tutoriel

Le tutoriel de la carte DE1-SoC Terasic (*Quartus_II_Introduction.pdf*) présente le logiciel Quartus, ainsi que les outils de saisie de modélisation, de simulation, et de synthèse de circuits configurables FPGA. Il permet de parcourir les différentes étapes de conception des circuits qui sont implémentés dans des FPGA. Le processus de conception est illustré en donnant des instructions pas à pas qui permettent d'implémenter un circuit très basique dans un dispositif d'Altera FPGA. Quartus II est un logiciel complet pour tous les types de modélisation, c'est-à-dire tous les types de langages de description (schématique, vhdl, verilog).

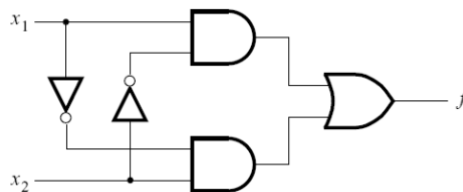


Figure 1 : Schéma de base à implémenter dans un FPGA

Pour cette initiation, il est demandé de décrire le système à partir d'une palette d'outils schématiques.

Remarque : Deux autres versions de ce cours d'instructions sont également disponibles, elles sont basées sur les langages de description de matériel Verilog et VHDL.

La dernière étape dans le processus de conception implique de configurer le circuit conçu dans un dispositif réel : pour la carte DE1-SoC, il s'agit d'un FPGA.

Les logiciels comme Quartus qui entrent dans la famille des CAD (Computer Aid Design) rendent facile l'implémentation d'un circuit logique dans un dispositif de logique programmable (PLD Programmable Logic Device), tel qu'une puce FPGA. Les différentes étapes à parcourir sont résumées dans le tableau ci-dessous :

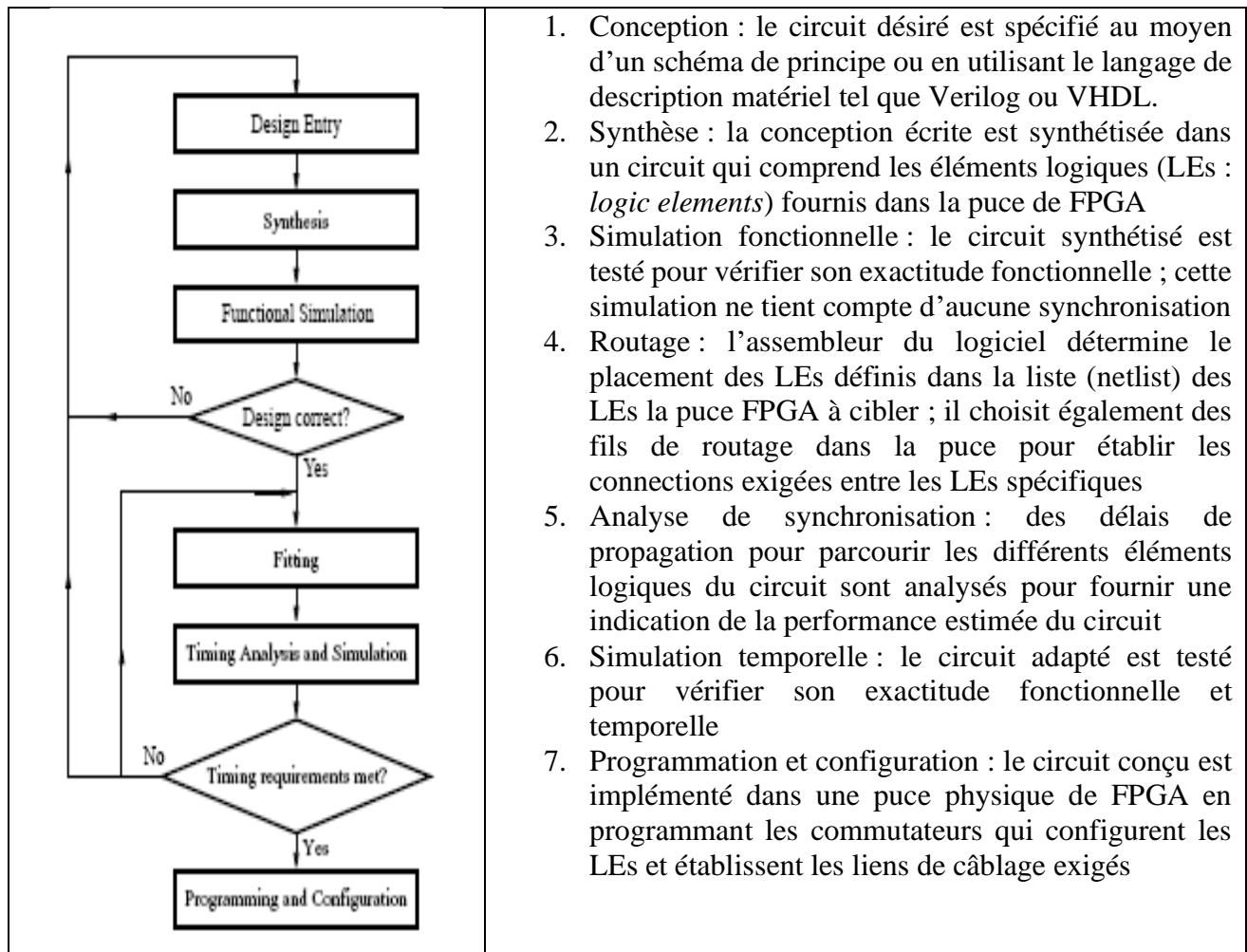


Figure 2 : étapes de conception d'un circuit

Fonctionnalité Quartus :

Les étapes décrites dans le tableau ci-dessus sont les bases de l'implémentation des PLD et elles sont indépendantes du logiciel de CAD (Computer Aid Design). Toutefois, selon les interfaçages, il est nécessaire de guider l'utilisateur dans les fonctionnalités du logiciel Quartus II. Le tutoriel vous guide donc également pour :

- La création d'un projet
- L'écriture d'un schéma de principe
- La synthèse logique
- L'identification d'un circuit synthétisé dans un FPGA de chez Altera
- L'affectation des entrées-sorties du circuit à des broches spécifiques du FPGA
- La simulation fonctionnelle et temporelle du circuit conçu
- La programmation et la configuration de la puce FPGA sur le panneau du DE1-SoC d'Altera

Partie II : Logique combinatoire à partir de l'éditeur graphique

L'objectif de la seconde partie de ce TP est de réaliser, toujours à partir d'une description schématique, les circuits décrivant un additionneur binaire complet, ainsi qu'un additionneur 4 bits qui sera synthétisé sur la carte d'expérimentation DE1-SoC.

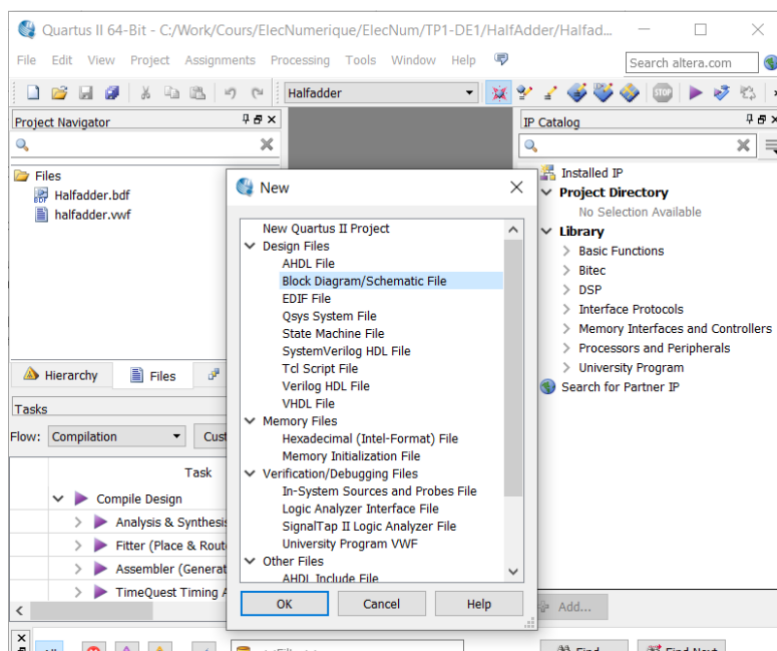
Le simulateur utilisé sera le simulateur intégré de Quartus : Qsim

Créer un Nouveau projet Quartus dans le répertoire « TP1/Additionneur », et nommé le par exemple : Additionneur.

Remarque : Cette étape est similaire à celle effectuée dans le tutoriel précédent concernant la création du projet quartus

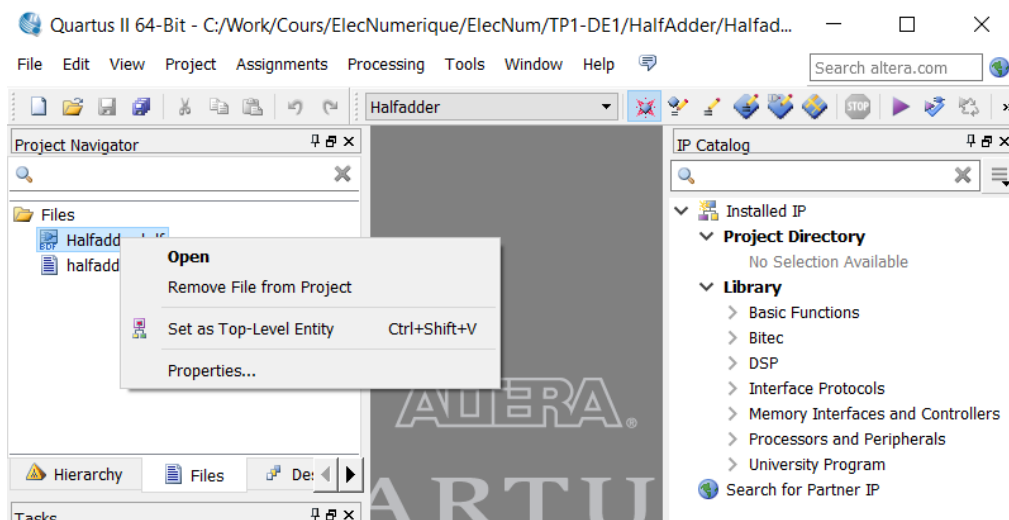
1. Semi Additionner 1 bits (2 opérandes d'entrées A et B et une sortie S qui fournit la somme de A+B et une retenue R)

- Créer un nouveau Bloc diagramme schématique
- Cliquer File / New -> Block diagram schematic



- Insérer les éléments logiques constituant un demi-additionneur.
- Compiler le projet
- Sauvegarder le schéma bloc avec le nom « Demi_Add »

- Dans la fenêtre Project Navigator, un clic droit pour faire apparaître le menu suivant et sélectionner : **Set as Top level entity**



- Effectuer une simulation fonctionnelle avec Qsim

2. Réaliser un Additionneur 1 bit complet :

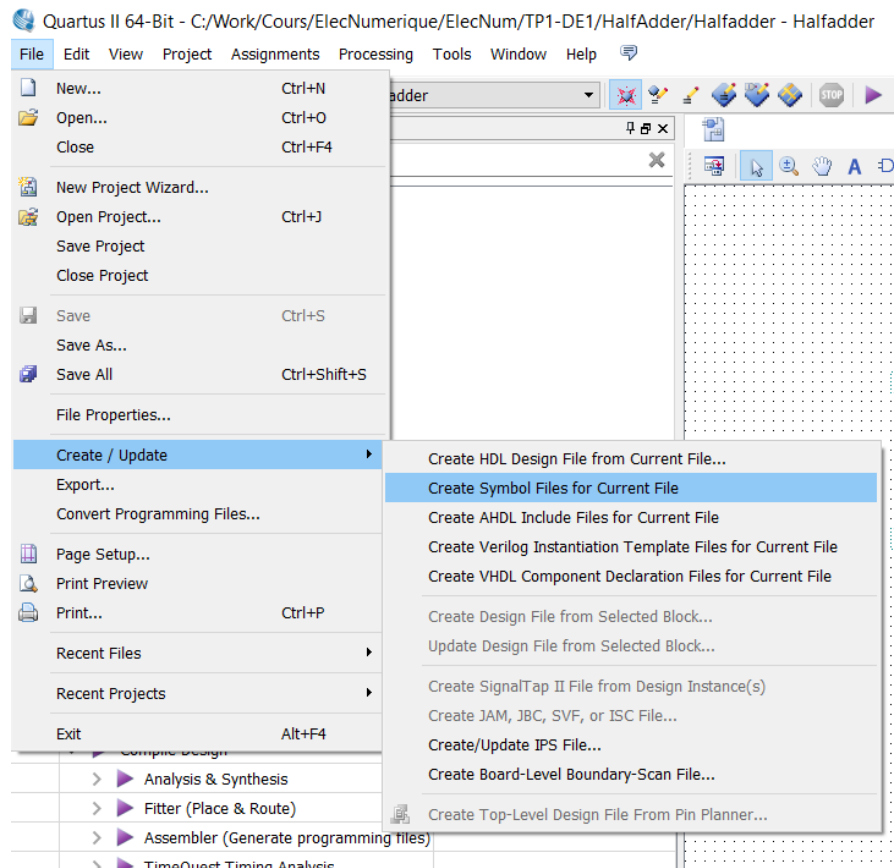
3 opérandes d'entrées : A et B et une retenue d'entrée Cin.

Une sortie S qui fournit la somme de A+B ainsi qu'une retenue de sortie Cout.

- Créer un nouveau Bloc diagramme schématique
- Insérer les éléments logiques constituant un additionneur complet
- Sauvegarder le schéma bloc avec le nom « Full_add_1bit.bdf»
- Dans la fenêtre Project Navigator, un clic droit pour faire apparaitre le menu déroulant et sélectionner : Set as Top level entity
- Effectuer une simulation fonctionnelle avec Qsim
- Synthétiser le circuit :

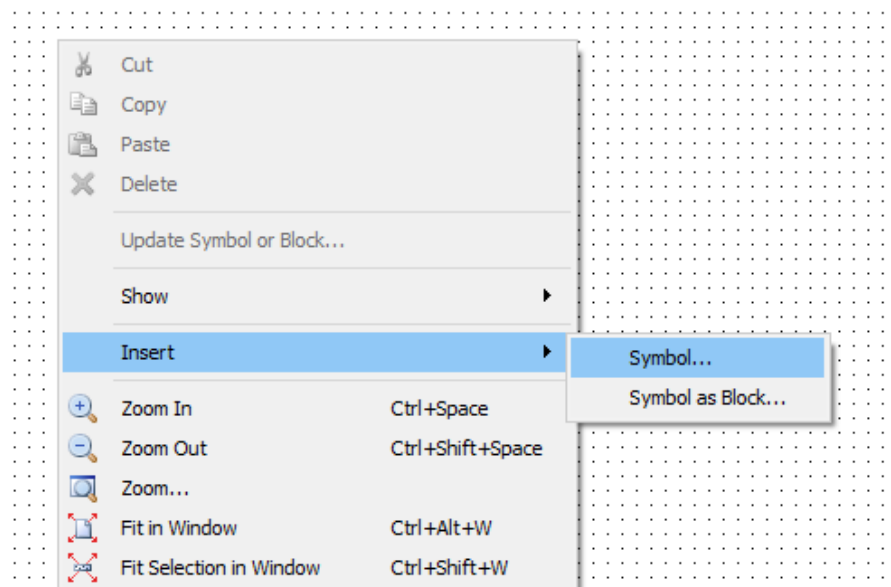
⇒ Création d'un Bloc Symbole Schématique

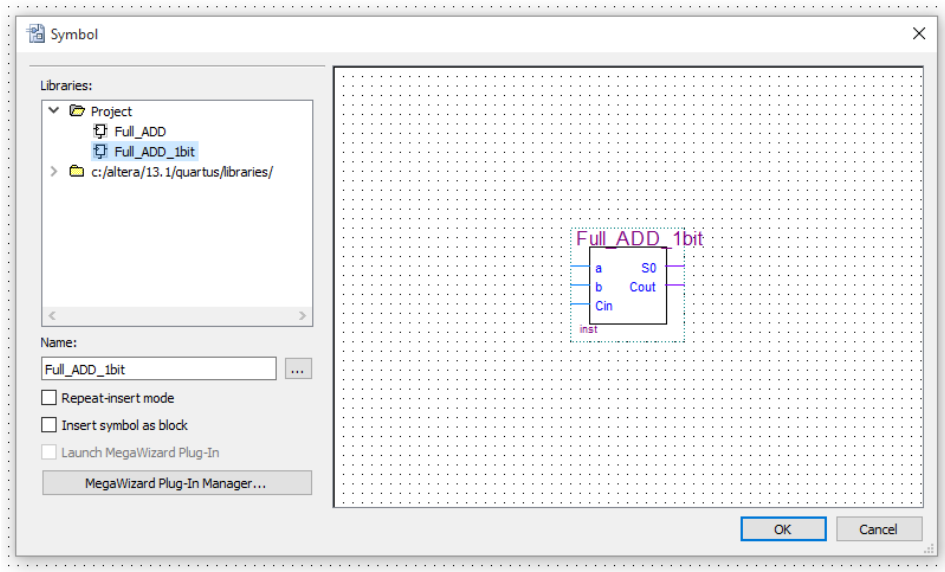
A partir de la description schématique de l'additionneur 1 bit complet que vous avez compilée et simulée dans l'étape précédente, procéder de la manière suivante pour la création d'un symbole Schématique. Il sera alors possible de réutiliser ce bloc fonctionnel pour un nouveau projet.



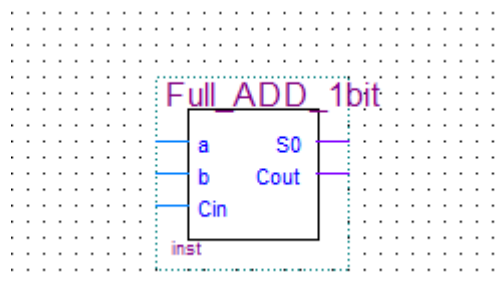
- Créer un nouveau Bloc diagramme Schématique et utiliser le symbole pour créer un additionneur 1 bit : *Full_add_1bit_top.bdf*
- Dans la fenêtre Project Navigator, un clic droit pour faire apparaître le menu déroulant et sélectionner : 'Set as Top level entity'
- Insérer le bloc fonctionnel : Full_add_1bit avec un

Clic droit sur la fiche diagramme :

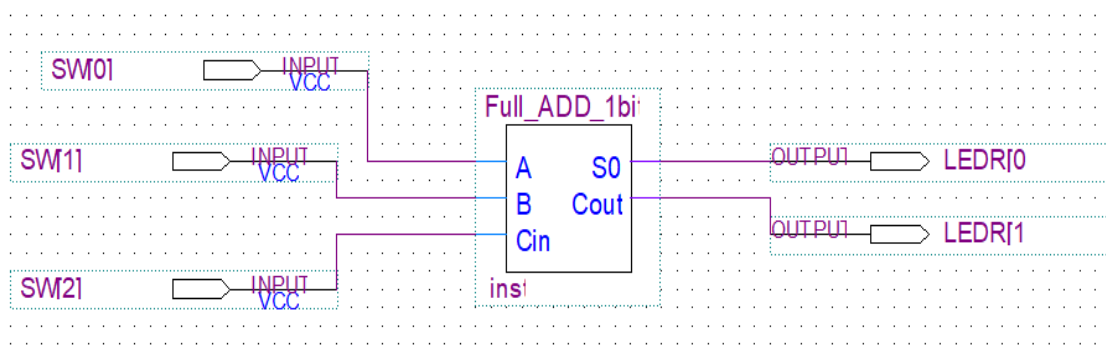




Cliquer ok et placer le bloc symbole sur le diagramme.



Insérer les pins input et output et nommer les, comme indiqué dans la figure ci-dessous.



- **Etape d'assignation des pins du circuit à la carte d'expérimentation DE1-SoC**

Cette étape de la synthèse de notre circuit peut être réalisée de différentes façons. Dans le tutoriel nous avons la méthode de base, à partir du menu :

Assignment > Assignment Editor

Ou alors

Assignment > Pin planer

Ces deux menus permettent d'associer les pins du FPGA aux signaux correspondant.

Ici, nous utilisons un fichier d'importation. Ce fichier associe l'ensemble des pins du FPGA avec des appellations de signaux prédéfinis. Ainsi, si l'on souhaite implémenter un design sur la carte DE1-SoC, il suffit d'utiliser le « *signal name* » en tant que nom du port d'entrée ou de sortie.

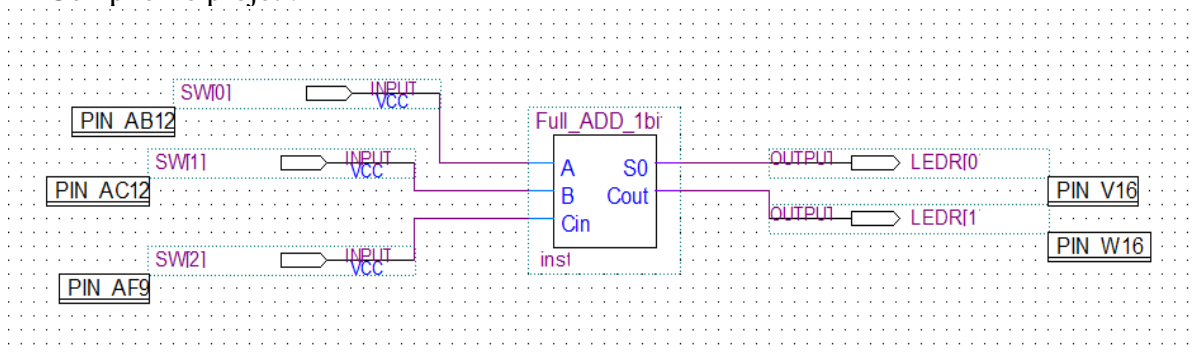
Ex : SW[1] correspond au switch 1 de la carte DE1-SoC.

L'utilisation de ce fichier d'importation facilite et permet de gagner du temps pour cette étape qui peut devenir assez fastidieuse pour des circuits plus complexes.

- Procéder comme suit pour l'importation du fichier d'assignation :

Importer le fichier '*DE1_SoC_Default.qsf*' qui se trouve sur le serveur

- Compiler le projet :



Les différents signaux en input ou output apparaissent alors assignés à un numéro de PIN du FPGA. Vous pouvez vérifier l'assignation dans l'outil PIN Planner

Avant de synthétiser le circuit dans le FPGA, il faut refaire une simulation fonctionnelle du circuit.

- Implémenter le design dans le FPGA

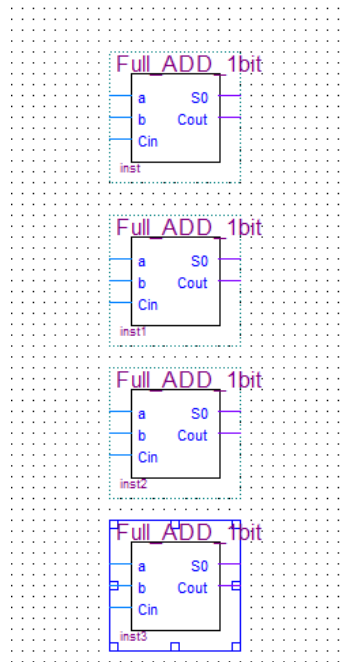
Menu :

Tools > Programmer

Vérifier que USB-Blaster est bien détecté puis lancer la programmation en appuyant sur Start
Vérifier le bon fonctionnement du circuit sur la carte DE1-SoC

3. Additionner 4 bits à partir de l'additionner 1 bit (Entrées : $A=A_3A_2A_1A_0$, $B=B_3B_2B_1B_0$, Cin. Sorties : $S=S_3S_2S_1S_0$ et Cout).

- Créer un nouveau Bloc diagramme schématique
- Sauvegarder le schéma bloc avec le nom « Full_add_4bit.bdf»
- Dans la fenêtre Project Navigator, un clic droit pour faire apparaître le menu déroulant et sélectionner : 'Set as Top level entity'
- Insérer 4 blocs fonctionnels 'Full_add_1bit' correspondant à l'additionneur complet 1 bit.



- Compléter le schéma pour réaliser un additionneur 4 bits

Il est demandé d'avoir recours à **des bus de données** pour concevoir les entrées A et B et la sortie S de ce circuit. A, B et S sont des demi octets chacun.

- Simuler le Circuit
- Implémenter sur la carte DE1-SoC

Fin du TP 1