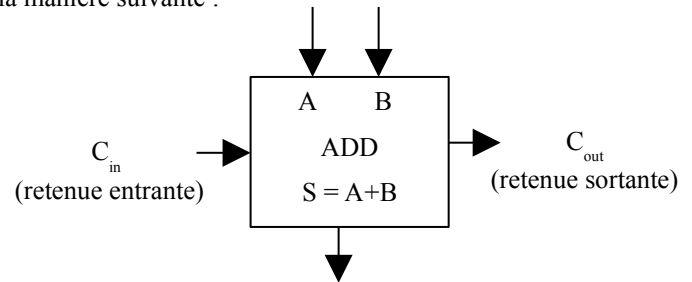


TP 1 – Additionneur

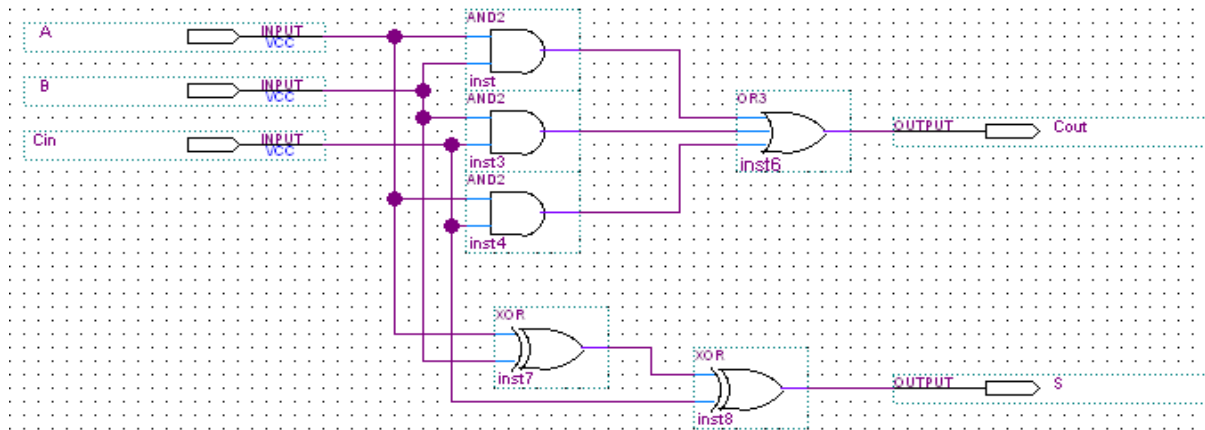
1. Travail préliminaire

Un additionneur un bit complet est un module logique qui effectue l'addition de 2 bits (A et B) et l'addition d'une retenue (C_{in}). Le résultat de cette addition est sur 2 bits que l'on décompose de la manière suivante :

- poids faible du résultat de somme (sur 1 bit) : S
- retenue propagée (poids fort du résultat de somme) : C_{out}



- Le logigramme est le suivant :



Vous donnerez les équations logiques correspondants aux sortie Cout et S.

2. Implantation sur circuit programmable

2.1. Saisie du schéma logique additionneur 1 bit sur logiciel

Dans cette partie vous allez saisir le logigramme de l'additionneur 1 bits complet, donné précédemment, dans le logiciel de développement Quartus II.

- Lancez le logiciel et créez un projet en utilisant le project wizard : sélectionner un répertoire personnel et rentrer le nom du projet (add par exemple) et enfin le nom de l'entité maître correspondant au fichier qui sera au sommet de la hiérarchie (top-level design entity).
- Ouvrez un nouveau fichier d'entrée : block diagram / schematic file que l'on sauvera en utilisant le nom du fichier maître (add1 par exemple)
- Saisissez votre schéma logique. Les composants sont appelés symboles, ils sont accessibles via des librairies. La librairie contenant les portes logiques est la librairie « primitives ». Les entrées et les sorties sont également des symboles (répertoire pin de la librairie primitives) le routage fonctionne comme avec un logiciel de dessin.
- Pour tester votre composant, il faut que le fichier fasse partie du projet et que ce fichier soit le 'top level entity'. Pour cela effectuez un clic droit sur le fichier sélectionné dans la fenêtre « project navigator » et cliquez sur « set as top-level entity »
- Pour vérifier que votre schéma logique est correct effectuez une compilation (menu « processing »). Le logiciel vérifie alors votre schéma, et en tire les équations logiques optimisées pour une synthèse logique. La synthèse logique correspondant à la réalisation du circuit logique dans un circuit intégré.
- Afin de réutiliser votre composant vous devez également générer un « symbole » graphique : menu File / create-update/ create symbole files.

2.2. Implantation dans un circuit intégré

Pour tester le bon fonctionnement du composant il est généralement préférable d'en effectuer la simulation. La simulation consiste à générer des entrées (0 ou 1) et à vérifier que l'état des sorties correspondent à ce qui est attendu.

Un logiciel de simulation est disponible pour des projets utilisant les langages de description VHDL ou Verilog mais pas pour des projets décrit à l'aide de logigrammes. Vous vous contenterez ici d'implanter directement l'additionneur 1 bit dans un circuit logique et de testez son bon fonctionnement sur les plaquettes de TP.

Pour l'implantation en circuit intégré, il est nécessaire de choisir le composant dans lequel va être implanté le circuit logique. Vous disposez en TP d'une platine sur laquelle est soudé un circuit FPGA d'Altera. Un circuit FPGA est un circuit intégré constitué par un réseau de portes logiques qui peuvent être interconnectées électroniquement entre elles. Lorsqu'un tel circuit est programmé certaines interconnexions sont coupées électroniquement de manière à réaliser le circuit logique désiré.

Le circuit FPGA est soudé sur un circuit imprimé (PCB) et les pattes du FPGA sont reliées électriquement à d'autres composants également soudés sur le circuit imprimé. On y trouve notamment des interrupteurs (switch) et des LED.

Pour programmer ce FPGA il faut :

- Sélectionner le bon composant parmi l'ensemble des circuits qui peuvent être programmés par le logiciel
- Affecter les entrées/sortie de votre projet aux pattes (PIN) du FPGA. (typiquement les entrées A, B et Cin seront affectées aux pattes du FPGA qui sont reliées aux switch et les sortie S et Cout seront affectées aux pattes du FPGA qui sont reliées aux LED)
- Compiler le projet
- Reliez le câble de programmation, alimenter la carte et programmer

1. La sélection du composant s'effectue à partir du menu « assignments → device ». Choisissez la famille MAX10 dans le menu déroulant (en haut à gauche de la fenêtre pop up). Sélectionner le numéro du composant dans la liste. Le numéro doit correspondre au numéro écrit sur la puce présente sur votre carte : 10M50DAF484C7G
2. Toujours dans la même fenêtre pop up, cliquez sur le bouton « Device and pin Options ». Sélectionnez « unused pins » et dans le menu déroulant sélectionnez « as input tri-stated ».
3. Pour l'affectation des entrées/sorties : « assignments → pin planner ». Vos entrées sorties apparaissent 'A, B, Cin, S, Cout). Cliquez deux fois dans les cases « location » pour faire apparaître un menu déroulant. Sélectionner le numéro de pin qui sera affecté à votre entrée/sortie. Répéter l'opération pour chaque entrée/sortie. Les connexions du FPGA aux composants de la carte sont données dans le user manual de la carte de10-lite.

Les numéros de pin à choisir sont :

- A (SW0) : PIN_C10
 - B (SW1) : PIN_C11
 - Cin (SW2) : PIN_D12
 - S(LED0) : PIN_A8
 - Cout (LED1) : PIN_A9
4. Compilez : « processing → compiler ». Cette opération va créer un fichier dit de programmation, qui correspond à l'activation ou désactivation de toutes les interconnexions du circuit FPGA en vue de réaliser votre projet. Cette opération peut être longue.
 5. Reliez le câble USB entre le PC et le connecteur « blaster » de la carte
 6. Programmez : « tools → programmer ». Sélectionnez pour le hardware le USBblaster (bouton Hardware Setup), cliquez sur Auto Detect et sélectionnez « 10M50DA », dans la partie haute double-cliquez sur le fichier <none> et sélectionnez le fichier .sof qui se trouve dans le répertoire output_file. Cliquez dans la case à cocher correspondant à Program/configure puis cliquez sur le bouton Start.
 7. Testez le bon fonctionnement de votre additionneur en envoyant plusieurs valeurs en entrée (switch) et en visualisant les valeurs des sorties (led)

2.3. Additionneur 4 bits

Saisie

Avant de procéder à la saisie du nouveau 'design', il est préférable de supprimer les affectations des entrées/sorties. Pour cela « Assignments→Remove Assignments → pin, location and Routing Assignments »

Ouvrez un nouveau fichier de saisie graphique afin de créer un additionneur 4 bits à partir de l'additionneur 1 bit réalisé précédemment. Pour récupérer votre composant procédez comme précédemment en allant le chercher dans la librairie correspondant à votre répertoire (il faut avoir créé le symbole auparavant, cf 2.1).

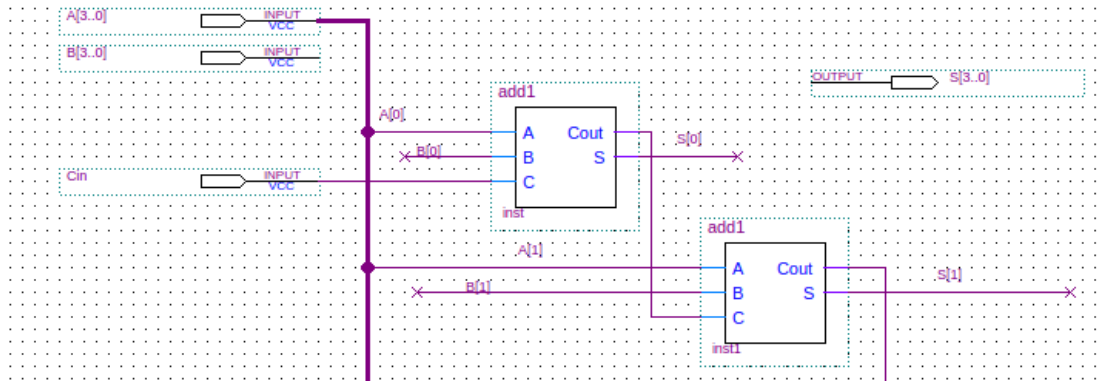
Votre nouveau fichier doit être désigné comme Top level entity (menu Project)

En entrée vous utiliserez deux bus : saisissez les entrées comme précédemment (input/output) en les nommant A[3..0] et B[3..0]. La sortie sera nommée S[3..0]. Les entrées de bloc étant sur un seul bit, il est nécessaire de donner un nom spécifiant quelle est la ligne (nod) du bus qui est utilisée : A[0], A[1], ... Pour nommer un fil ou bus faire un clic droit sur le fil et « properties » ; name : le nom que vous souhaitez donner.

Les bus doivent être en gras « bus line » et les simples lignes (node) non. Les lignes peuvent être reliées physiquement au bus mais cela n'est qu'optionnel. **La connexion s'effectue par nom.** Les bus doivent donc avoir des noms (excepté dans les cas des entrées sorties comme ici où le bus prend le nom de l'A/S). Voir schéma ci-après (Attention le schéma n'est pas complet!)

Pour réaliser l'additionneur 4 bit il est nécessaire de disposer de 4 additionneur 1 bits. L'addition s'effectue du poids faible (A[0], B[0]) vers le poids fort. La retenue se propage du poids faible vers le poids fort.

Réalisez le schéma de l'additionneur 4 bits, sélectionnez votre schéma comme « top level entity » et lancer la vérification logicielle



Implantation

Affectez les entrées aux pattes du FPGA reliées aux switches et les sorties aux pattes du FPGA reliées aux LED. Se référer au User manual.

Lancer la compilation et implanter votre projet dans le FPGA

Testez

3. Additionneur/soustracteur (Pour les + rapides)

A l'aide de l'additionneur précédemment réalisé, on souhaite réaliser un additionneur/soustracteur.

- La soustraction d'un nombre par un autre sur un format fixe (ici 4 bits) s'effectue par l'addition du complément à 2 du nombre à soustraire. i.e : $A - B$ est réalisé par $A + \bar{B} + 1$, Ou \bar{B} correspond à tous les bit complémentés.
- Réalisez un additionneur/soustracteur 4 bits sans retenue d'entrée mais avec une retenue de sortie. Une entrée supplémentaire est ajoutée pour sélectionner entre l'addition et la soustraction :



- Aide : lorsque $add_sub = 1$, il faut que sur les entrées de l'additionneur 4 bits réalisé au 2. que l'on renommera A_p , B_p et Cin_p on ait A , B et 0 lorsque $add_sub = 1$, et A , \bar{B} et 1 lorsque $add_sub = 0$. Donnez les équations de B_p fonction de B et add_cin et Cin_p fonction de add_sub .
- Réaliser le schéma, et implantez le dans le FPGA