

## TD n°3 : Réalisation d'instructions SIMD pour le processeur NIOS II

### 0. Introduction

On peut définir de nouvelles instructions (« customization ») pour le processeur NIOS II (cœur logiciel pour FPGA d'Altera), selon le schéma

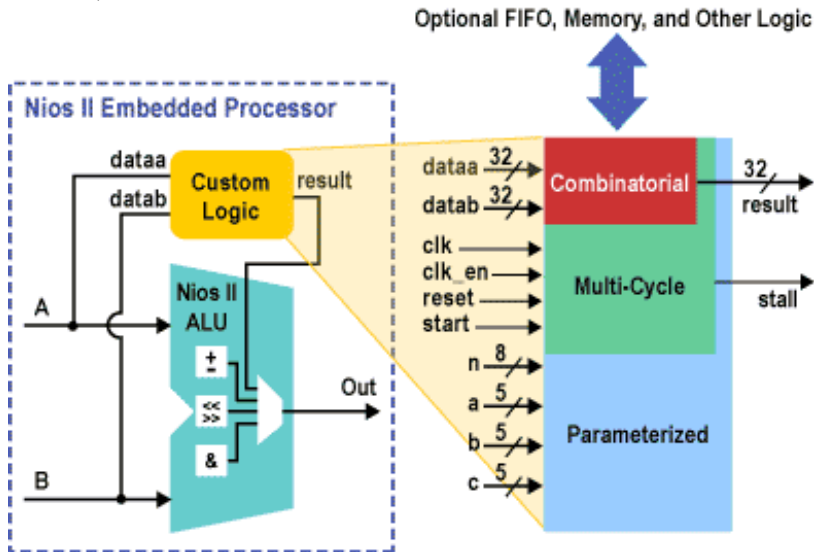


Figure 1 : Spécialisation d'instructions pour le processeurs NIOS II

### 1. Opérateurs MIN

#### Opérateur min

L'opérateur min est implanté en C à l'aide d'une fonction. Dans une première étape, on veut réaliser un opérateur MIN sur des entiers non signés 32 bits. Ecrire le code VHDL permettant d'implanter le matériel pour une instruction spécialisée de ce type.

#### Filtre MIN 3 x 3 pour le traitement d'images.

On veut réaliser les opérateurs matériels pour des instructions SIMD permettant de traiter simultanément 4 octets non signés (pixels) pour le filtre MIN. Les instructions à définir sont

- Décalage gauche d'un octet pour obtenir le mot de 32 bits pour les pixels  $X[i][j-1]$ .
- Décalage droite d'un octet pour obtenir le mot de 32 bits pour les pixels  $X[i][j+1]$
- Instruction SIMD de minimum sur des octets non signés.

Ecrire le code VHDL pour l'implantation de ces différentes instructions.

### 2. Gauss

Sur un tableau  $N \times N$  pixels (niveau de gris), on applique le filtre de Gauss 3 x 3 ci-dessous

$$1/16 * \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

On veut réaliser les opérateurs matériels pour des instructions SIMD sur 32 bits permettant d'accélérer l'application du filtre de Gauss.

Les instructions à définir sont

- Décalages à gauche et droite d'un octet de mots de 32 bits (déjà définis dans la question précédente)
- Décompactage de deux octets bas en deux demi-mots de 16 bits (extension de zéros)
- Décompactage de deux octets haut en deux demi-mots de 16 bits (extension de zéros)
- Décalage gauche d'une et deux positions des demi-mots d'un mot de 32 bits

- Addition SIMD de deux demi-mots non signés.
- Compactage de deux demi-mots d'un premier mot en deux octets bas du mot résultat et des deux octets haut d'un deuxième mot en deux octets hauts du mot résultat. On considérera deux versions : la première considère que les demi-mots (16 bits) ont leur octet haut nul ; la seconde version sature l'octet bas lorsque l'octet haut du demi-mot est différent de 0.

Définir le code VHDL pour ces différentes instructions.

### **3. Filtre médian**

Définir le code VHDL pour les instructions nécessaires à l'implantation du filtre médian.