|  |
| --- |
| Rapport du laboratoire 4 d’INF3610 – Hiver 2018 |
| NOËLÉ Elodie & THIBAULT Maxime |
|  |
| 1796174 – 1932759 |
|  |

**Questions initiales :**

**Latence de boucles:**

***Etape 1 :***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Loop name** | **min** | **max** | **Iteration latency** | **Achieved** | **Target** | **Trip count** | **Pipelined** |
| **IMG** | 165888001 | 165888002 | 8 | - | - | 2073600 | no |
| **OneTo4** | 4 | 4 | 1 | - | - | 4 | no |

***Etape 2 :***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Loop name** | **min** | **max** | **Iteration latency** | **Achieved** | **Target** | **Trip count** | **Pipelined** |
| **IMG** | 6220800 | 6220800 | 3 | - | - | 2073600 | no |

***Etape 3 : #pragma HLS unroll factor = 8***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Loop name** | **min** | **max** | **Iteration latency** | **Achieved** | **Target** | **Trip count** | **Pipelined** |
| **IMG** | 12960000 | 12960000 | 50 | - | - | 259200 | no |
| **OneTo4 (répété huit fois)** | 4 | 4 | 1 | - | - | 4 | no |

***Etape 3 : #pragma HLS pipeline OneTo4***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Loop name** | **min** | **max** | **Iteration latency** | **Achieved** | **Target** | **Trip count** | **Pipelined** |
| **IMG\_OneTo4** | 8294402 | 8294402 | 4 | 1 | 1 | 8294400 | yes |

***Etape 3 : #pragma HLS pipeline IMG***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Loop name** | **min** | **max** | **Iteration latency** | **Achieved** | **Target** | **Trip count** | **Pipelined** |
| **IMG** | 2073601 | 2073601 | 3 | 1 | 1 | 2073600 | yes |

**Question 1. Vous avez un code impliquant une convolution 2D 3x3 (soit une multiplication-accumulation sur 2 boucles imbriquées à 3 itérations/boucle, comme *sobel\_operator*) à accélérer par HLS. Cette convolution est exécutée assez souvent (assez pour qu’une optimisation nous apporte un gain important), mais pas de manière régulière (ex. pas dans une plus grosse boucle sans contrôle9). En vous basant sur les résultats précédents, devriez-vous dérouler ou pipeliner ces deux boucles? Expliquez pourquoi votre choix est le meilleur, et pourquoi l’autre choix est moins bon (autrement dit, donnez l’avantage de votre choix dans cette situation, et le désavantage de l’autre choix dans cette situation).**

Il serait préférable de dérouler les deux boucles. En effet, si l’on compare #pragma HLS unroll avec #pragma HLS pipeline OneTo4, on peut noter une baisse du nombre d’itérations réalisées par la boucle IMG. En outre, la boucle OneTo4 a une latence de 4 lorsque l’opération est pipelinée, alors que cette latence n’est que de 1 lorsque l’on déroule les deux boucles. ( A compléter ?)

**Question 2. Vous avez un code impliquant une convolution 2D 3x3 à accélérer, mais celle-ci est appelée régulièrement et sans contrôle significatif en boucle pour couvrir une image entière de résolution appréciable. En vous basant sur les résultats précédents, devriez-vous dérouler ou pipeliner cette boucle principale ? Expliquez pourquoi votre choix est le meilleur, et pourquoi l’autre choix est moins bon (autrement dit, donnez l’avantage de votre choix dans cette situation, et le désavantage de l’autre choix dans cette situation).**

Il est préférable de pipeliner la boucle principale. Si l’on compare #pragma HLS unroll avec #pragma HLS pipeline IMG, on peut noter une baisse du nombre d’itérations et de la latence de la boucle IMG. . ( A compléter ?)

**Questions supplémentaires**

**Analyse/Question 3a.** Remettez le tableau contenant la performance de toutes vos itérations et annotez-le des modifications que vous avez fait entre chaque itération, expliquant la différence de performance.

**Question 3b.** Pour chacune de vos itérations, comparez la performance prévue par Vivado HLS (nombre de cycles x période d’horloge) et celle obtenue en réalité. Les deux valeurs sont-elles similaires? Assez différentes? Différentes dans certains cas et similaires pour d’autres? Expliquez ces résultats.

**Question 4**. On dit que les architectures RISC de type superscalaire et multicoeurs sont utiles pour une parallélisation à gros grains alors que le FPGA est utile pour une parallélisation à grains fins. Illustrez cette affirmation à l’aide du travail effectué dans ce laboratoire. Autrement dit, aurait-on pu obtenir les accélérations obtenues dans ce laboratoire sur vous eut accès à une architecture 9 coeurs? Expliquez.

**Question 5.** Quelle différence existe-t-il entre *#pragma HLS pipeline et du pipelinage logiciel (que nous verrons en classe au prochain cours)? Expliquez.*