

Intel HLS Compiler を用いた 画像処理ハードウェア設計に関する検討

宮地 航太[†] 渡邊 誠也[†] 名古屋 彰[†]

[†] 岡山大学大学院自然科学研究科

1. はじめに

近年、画像認識等の分野において、ハードウェアアクセラレータを用いたシステムが注目されている。しかし、ハードウェア設計では、一般にハードウェア記述言語を用いる必要があり、開発が困難となっている。そこで、C 言語等の高水準言語から RTL 記述を生成する高位合成技術が注目されている。しかし、例えば Intel 社が開発した高位合成ツール Intel HLS Compiler[1]はリリースされたばかりであることから、記述と合成されるハードウェアとの関係は明らかでなく、設計者の求めるハードウェアが高位合成によって得られるとは限らない。

そこで、本研究では高位合成ツール Intel HLS Compiler を用いてアプリケーションを実装し、ツールが合成するハードウェアの構成や性能について明確化し、ツールを用いる上での指針を得ることを目的とする。

2. 高位合成ツール Intel HLS Compiler

Intel HLS Compiler は、Intel 社が提供する C++言語を RTL 記述に変換する高位合成ツールである。

図 1 に高位合成ツールを用いた実装の流れを示す。通常、RTL 記述よりも C++記述は高速に動作の検証ができるため、その時間を大幅に短縮することができる。

3. 研究手法

本研究では、高位合成ツールを用いて様々な画像処理アプリケーションの実装を行い、合成されるハードウェアの構成や性能を評価する。また、同様の動作を行うハードウェアを RTL でハンドコーディングした実装も行い、高位合成ツールが生成したハードウェアとの性能を比較する。まず、エッジ検出で用いられるソーベルフィルタの実装を行った。

4. 実装と評価

ハードウェアの性能の評価には Intel 社の論理合成ツール Quartus Prime を用い、実装対象の FPGA は Cyclone V SoC の 5CSXFC6D6F31C6 とする。表1に各実装のハードウェアを論理合成し、得られた結果を示す。表中のスループットは最大動作周波数で 1 秒間に処理することができる

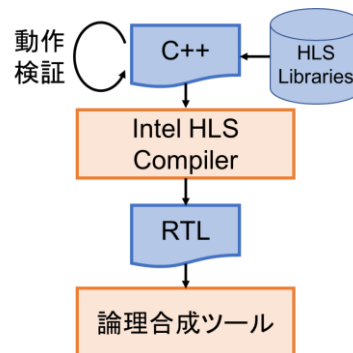


図1. 実装の流れ

Full HD (1920×1080)画像のフレーム数である。ここで、高位合成する C++のコードには高位合成におけるハードウェア最適化のための特別な記述は追加していない。また、ハンドコーディングしたハードウェアはパイプライン化していない。

ハンドコーディングしたハードウェアの方がリソース量が少なく、スループットも高いことが分かる。特に、ALM 数は高位合成ツールのハードウェアが約 9.5 倍になっている。原因は、ソーベルフィルタ処理における平方根の演算がリソースを多く消費しているからである。ハンドコーディングしたハードウェアはビット幅に適した平方根の演算器を用いているのに対し、高位合成ツールでは余分なビット幅の演算器を生成し、多くのリソースを使用したと考える。

高位合成したハードウェアはパイプライン構成となっているが詳細なパイプラインの構成を十分に分析していないため、更なる調査を進める。

5. 今後の課題

今後の課題として、合成されたハードウェアに対して詳しく調査を行い、ハードウェアの構成を明確化することや高位合成ツールにおけるハードウェアの最適化により構成がどのように変化するのかを明確化することがあげられる。また、他のアプリケーションの実装と評価も進める。

参考文献

- [1] <https://www.altera.co.jp/products/design-software/high-level-design/intel-hls-compiler/overview.html>

表1. 論理合成結果

	使用ALM数	使用メモリビット数 (bit)	使用レジスタ数	使用DSP Block数	最大動作 周波数 (MHz)	スループット (フレーム/秒)
Intel HLS Compiler	1,840	91,408	5,209	18	203.9	98
Hand coding	191	32,768	204	1	243.7	117