

D7 5/3 ウェーブレット変換のためのインプレース演算モジュール

水谷 直喜 三上 喬 村松 正吾 佐々木 重信 周杰 菊池 久和

新潟大学工学部電気電子工学科

1. はじめに

近年標準化された静止画像の国際標準である JPEG2000 は、離散ウェーブレット変換 (Discrete Wavelet Transform; DWT) を採用しており、この実現法としてリフティングを定義している[1]. このリフティング演算法はインプレース演算が可能であることが知られている。このインプレース演算は、演算結果を格納するための別のバッファを用意することがないので使用メモリが少ないという利点をもつ。また、現在とその前後のサンプルのみの簡便なフィルタによって実現される。

一般に画像処理などの応用分野においては、演算量が膨大となるため、演算速度向上などの目的のために高速なハードウェア実装が望まれる。高速実装には演算量と同時にメモリ転送量の低減が有効である。メモリ転送の点では、メモリ自体に演算を組み込む機能メモリ化が有効な方法である。リフティングのインプレース演算はこのような実装に適している。一方、面積を重視した場合、演算部の資源を時分割で共有したい場合もある。このような場合においても、インプレース演算はその局所性からメモリ転送の点で有効である。本報告ではこのインプレース演算の特徴を生かすために、一つのセルに演算機能と記憶素子の両方を合わせ持つインプレース演算モジュールを提案する。このモジュール化により様々な設計要求に対して、このモジュールの接続を換えるだけで対応することができる。

2. インプレース演算によるリフティング構成

2-1 リフティング構成

図1にリフティング構成を示す。JPEG2000 Part1では(5,3)フィルタと(9,7)フィルタが採用されている。本報告では(5,3)フィルタにおけるリフティング演算について検討を行う。

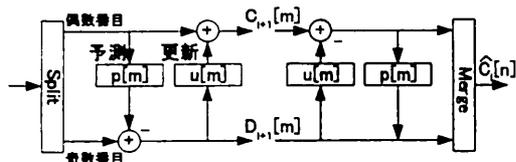


図1 リフティング構成

リフティングは、入力信号の偶数番目と奇数番目を分離する Split, 予測フィルタ $P[m]$, 更新フィルタ $U[m]$ 及び奇数番目と偶数番目のサンプルを合成する Merge から構成される。なお、(5,3)フィルタの場合、予測フィルタの係数は(1/2, 1/2), 更新フィルタの係数は(1/4, 1/4)である。図1の $C_{i+1}[m]$ は低周波係数, $D_{i+1}[m]$ は高周波係数を示す。リフティング演算は、低演算量、整数演算で完全再構成が可能であり、インプレース演算によりメモリ使用量が少ないという利点を持っている。次節ではインプレース演算について説明する。

2-2 インプレース演算

図2に入力係数のサンプル数が4点の場合のインプレース演算処理過程を示す。なお、実線はサンプルを1倍、点線は-1/2倍、1点鎖線は1/4倍することを示す。また、図中の番号はサンプルのインデックスを示す。また、対称拡張を仮定している。

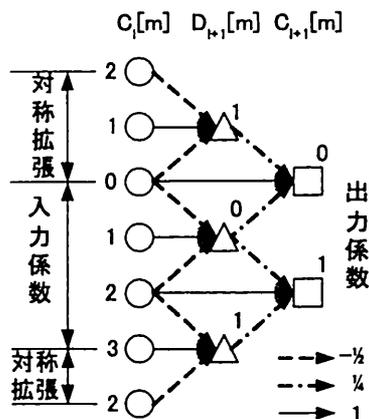


図2 インプレース演算の流れ

以下に図2の流れを説明する。

- 0番目と2番目のサンプルをそれぞれ-1/2倍し、これらを1番目のサンプルと足し合わせ、 $D_{i+1}[m]$ の1つ目のサンプルを出力
- 入力係数の1番目のサンプルを今計算されたサンプルと置き換える
- 同様に全ての $D_{i+1}[m]$ の係数を計算

(1番目の $D_{i+1}[m]$ の計算時は入力信号の境界のため計算に必要なサンプルが足りない所以对称拡張する)

- $D_{i+1}[m]$ の0番目と対称拡張された1番目の係数を $-1/4$ 倍し, 入力係数の0番目サンプルとを足し合わせて $C_{i+1}[m]$ の1つ目のサンプルを出力
- 入力信号の0番目のサンプルを今計算されたサンプルと置き換える
- 同様に全ての $C_{i+1}[m]$ のサンプルを計算

以上のことから, インプレース演算によりバッファなどを用意しなくてもリフティング演算が可能であるので, 使用メモリが少ないことが分かる。

3. インプレース演算モジュールの提案

図3に本報告で提案するインプレース演算モジュールを示す。図中の点線はコントロール信号を示す。

以下に図3の特徴について述べる。

- 現在の入力サンプルを DFF に記憶
- その前後のサンプルを入力より読み込み, コントロール信号により, 符号(+ or -)と乗算する係数(1/2 もしくは 1/4)を切り替えながらインプレース演算を行う (乗算を行う部分はシフト演算により構成される)

インプレース演算モジュールの応用例として, 図2に示す4点DWTを4つの提案モジュールでインプレース演算実装した例を示す。(図4)

以下に図4の手順を示す。

- 入力サンプルを各CellのDFFに記憶 (CONTROL 信号にインプレースモード'0'が入力され各DFFへのデータセットが行われる)
- $D_{i+1}[m]$ を計算する時は Cell_1 と Cell_3 の EN に, $C_{i+1}[m]$ を計算する時は Cell_0 と Cell_2 の EN に'1'を入力して対応する各モジュールでは演算が行われる
- 演算が終了すると必要に応じて, 出力部より計算されたサンプルを取り出す

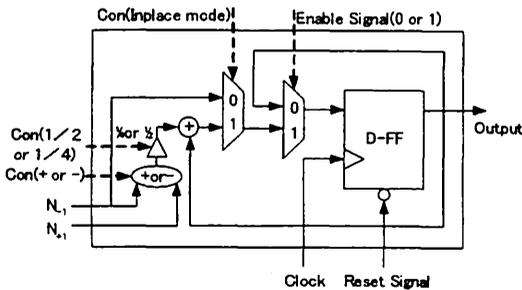


図3 インプレース演算モジュール

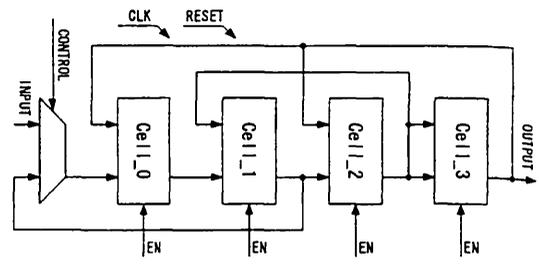


図4 インプレース演算モジュール直接構成図

4. シミュレーションと合成結果

4-1 シミュレーション

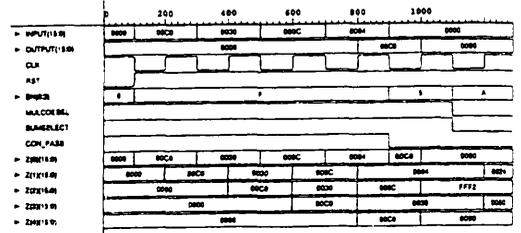


図5 図4の直接構成のシミュレーション波形

MULCOESEL は係数選択信号, SUMSELECT は符号の選択信号, CON_PASS はコントロール信号を示す。

4-2 合成結果

図4の直接構成をVHDLでモデル化し, 合成結果から性能評価を行った。合成にはXILINX社のSPARTAN 2 XC2S200-5PQ208をターゲットデバイスとした。合成ツールとしてXILINX社のISE4.2iを用いた。合成結果より等価ゲート数は2591であった。

5. おわりに

本報告では, 5/3 ウェーブレット変換のためのインプレース演算モジュールを提案した。また, その直接構成の合成及びシミュレーションを行い, モジュール化の有効性を確認した。

謝辞

本研究の一部は, 日本学術振興会科学研究費若手研究(B)No.14750283の助成の下で行われた。

参考文献

[1] David S. Taubman, Michael W. Marcellin "JPEG2000", Kluwer Academic Publishers 2002
 [2] Uwe Meyer-Baese "Digital Signal Processing with Field Programmable Gate Arrays", Springer-Verlag Berlin Heidelberg 2001