

2次元シストリックアレイ並列計算モデルの記述とモデル検査器を用いた振る舞い検証

Verification and Description Using a Model Checker for 2D Systolic Array Parallel Models

千葉 悠矢 † 和崎 克己 ††
Yuya Chiba Katsumi Wasaki

1 はじめに

ハードウェア設計においてハードウェア回路検証は実際のハードウェア製造のコストが大きいと、開発コスト削減の面で重要な要素である。ハードウェア回路をモデル化し、モデル検査を行う形式手法も動作検証の1つとして用いられる。

本研究では2次元シストリックアレイの LOTOS 言語による記述を行い、CADP toolbox[1]を用いて、生成したモデルの振る舞いを検証する。モデル化では、グローバルクロックによる同期型により構成する。同期型構成によりシストリックアレイの抽象度を下げ、それぞれの素子の振る舞いを観測可能とすることで、局所的なメッセージ非同期性とグローバル同期性の両方を検査する。

2 シストリックアレイにおける並列計算モデル

シストリックアレイは大量の単純な処理を行う Process Element(PE)を1次元または2次元上配置し、一度の入力により、生成される計算途中の一時データをメモリに書き込むことなく、演算を行うよう構成されたアーキテクチャである [2]。加算や乗算などの単純な処理を行う PE を複数個用意し、PE の入力と出力を PE 同士で接続し、規則的に配置することにより、システム全体として行列積計算などが可能である。PE の配置や PE 単体の内部演算を変更することで様々な計算が可能である。単純な構成法として、同期回路によりシストリックアレイを構成する。それぞれの PE の入出力すべてを単一のグローバルクロックにより同期しながら処理を行う必要がある。

本研究では2次元シストリックアレイを対象としてモデル化を行う。2次元シストリックアレイはセルに保持された重みの値と入力列を各 PE 内で計算を行い次の PE に出力する。システム全体として行列積を計算することができる。図1に2次元シストリックアレイにおけるセル単体の仕様を示す。2次元シストリックアレイでは上部から下部へと入力列のストリームが流れており、上部からの入力は下部のセルへと伝搬される。右方向のストリームでは PE での計算結果が伝搬される。左側からの入力と上部から入力により PE 内部で計算を行い、右側へ計算結果の出力を行う。作成したモデルの PE 単体の仕様として上部からの入力 x と PE があらかじめ保持している重み W を乗算し、右方向ストリームからの

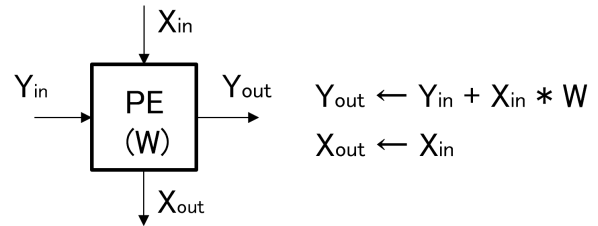


図1 2次元型におけるセル単体の仕様

値と加算した結果を出力する。PE を正方形に配置することで、行列積を計算するシストリックアレイを構成する。

3 LOTOS 記述

シストリックアレイをモデル化するにあたり、PE 内部の加算器、乗算器、レジスタの3つの構成要素をプロセスとして扱い、3つの構成要素を組み合わせ、各プロセスを合成することにより、PE 単体をプロセスとして生成する。PE プロセスに対してそれぞれ重み W を初期値として与え、シストリックアレイと同様に配置することにより、シストリックアレイ全体のプロセスを作成する。

3.1 グローバルクロックを用いた LOTOS 記述

グローバルクロック同期型2次元シストリックアレイの構成要素として加算器、乗算器、レジスタがある。これらをプロセスとして LOTOS 上で定義し、それぞれのプロセスの組み合わせによって PE 単体を記述する。加算器1つ、乗算器1つ、レジスタ2つで1つの PE を構成し、複数のセルを用意して互いに接続することでシストリックアレイモデルを作成する [2]。図2に同期回路における構成図、図3に PE 単体の構成を示す。また図4に図2に対する LOTOS 記述、図5に図3に対する LOTOS 記述を示す。

グローバルクロックと入力を制御するプロセス GENERATOR(GEN)を定義し、GEN が適切なタイミングで入力列とクロック入力を行うように定める。

4 振る舞い検証

本モデルでは PE を4つ用意し、入力列 $X_{11}, X_{12}, X_{21}, X_{22}$ 、重み列 $W_{11}, W_{12}, W_{21}, W_{22}$ を与える。重み列は PE 内部にあらかじめプリセットしておく。入力列が上部から伝搬するときのシストリックアレイ全体の振る舞いをモデル検査器である CADP toolbox を用いて振る舞いを示したグラフである Labelled

† 信州大学大学院総合理工学研究科, Graduate School of Science and Technology, Shinshu University

†† 信州大学工学部電子情報システム工学科, Department of Electrical and Computer Engineering, Faculty of Engineering, Shinshu University

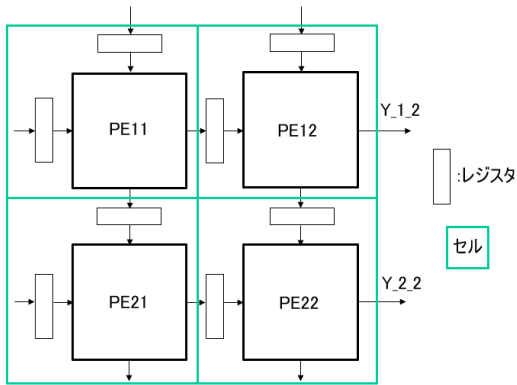


図2 2次元型シストリックアレイの同期回路構成図

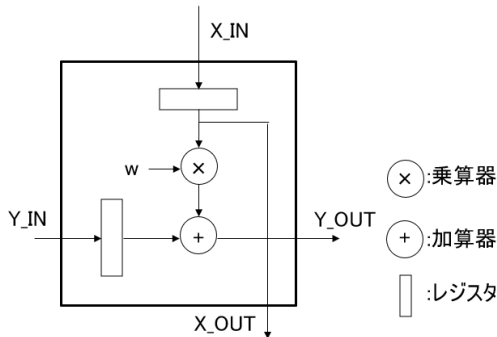


図3 PE単体の構成図

Transition System(LTS) を出力する。出力した LTS を検査することによりモデルがどのような振る舞いが現れるか検証することができる。

PE 内部の加算器及び乗算器の出力の遅延により、仕様と異なる計算結果になる振る舞いが存在する。ここで加算器と乗算器の遅延が存在しない振る舞いのみを抽出した LTS を生成する。遅延が存在しない LTS の生成にはプロセスの出力アクションに優先度を用いる。具体的には各 PE すべてを同期するグローバルクロックの出力が最も優先度が低くなるように定める。優先度の適用によって加算器及び乗算器がすでに完了した状態からグローバルクロックによりすべてレジスタが同期され、次のクロックサイクルに状態が移行する。図6に各PE内部での計算結果を出力するを右方向出力信号線のみを抽出した LTS を示す。優先度を用いて生成された LTS においてすべての振る舞いの中で出力はすべて仕様を満足することが確認できた。

5 まとめと今後の課題

本研究では2次元シストリックアレイにおける LOTOS 記述と振る舞い検証を行った。今後の課題として検証事項の自動化がある。シストリックアレイの構成における繰り返し構造に着目した検証事項の確立や繰り返し現れるような状態遷移のパターンマッチによりプログラムによる検証事項の自動生成を目的とする。

```

...
process Array[Y_0_2,Y_0_1,...,CLK] : noexit :=
  (Cell[X_1_0,X_1_1,Y_0_1,Y_1_1,...,CLK](W11)
   |[X_1_1,CLK]|
   Cell[X_1_1,X_1_2,Y_0_2,Y_1_2,...,CLK](W21)
   |[Y_1_1,Y_1_2,CLK]|
   (Cell[X_2_0,X_2_1,Y_1_1,Y_2_1,...,CLK](W12)
    |[X_2_1,CLK]|
    Cell[X_2_1,X_2_2,Y_1_2,Y_2_2,...,CLK](W22)
   endproc(*Cell*)
...

```

図4 シストリックアレイ構成の LOTOS 記述 (全体構成)

```

...
process Cell[X_IN,X_OUT,Y_IN,Y_OUT,...,CLK](W:EXP)
  : noexit :=
  (DFlipFlop_EXP[Y_IN,CLK,A_IN](..., (0 of Bit))
   |[CLK]|
   DFlipFlop_EXP[X_IN,CLK,X_OUT](..., (0 of Bit)))
   |[A_IN,X_OUT]|
  (Adder_EXP[M_OUT,A_IN,Y_OUT]
   |[M_OUT]|
   Multiplier_EXP[X_OUT,M_OUT](W)
  )
endproc(*Cell*)
...

```

図5 シストリックアレイ構成の LOTOS 記述 (セル単体)

