モジュール間転送時間を考慮した ロジックインメモリVLSIシステムのハイレベルシンセシス

1H−2

堀井 崇史　羽生 貴弘　亀山 充隆
東北大学大学院情報科学研究科

1．まえがき

VLSI技術の進展に伴い，能動素子の高性能化が達成される一方，配線量の増大に伴う性能劣化が問題となってきている．このため配線量や配線遅延を考慮した，ハイレベルシンセシスの必要性が高まっている [1]．本稿では，粒度の小さい演算器やメモリ，レジスタという基本モジュールを用い，それらの物理レベルのレイアウトを考慮して，データ転送時間を求め，システム全体の処理時間を最小にする，ロジックインメモリVLSIシステムのアロケーション法について述べる．すなわち，1ステップにかかる処理時間である，基本モジュール間のデータ転送時間と演算時間の和の最大値を最小にすることで，システム全体の処理時間を最小にするアロケーション法について提案する．

2．ハードウェアモデル

入力として図1のようなスケジューリングされたデータフローグラフ (SDFG) を考える．このSDFGでは記憶もノードとして表し，アークはデータ転送のみを表す．このSDFG中の，ある1組の依存関係にあるノードに着目すると，1ステップにかかる処理時間は，主にメモリやレジスタから演算器までのデータ転送時間と演算器内部の演算時間の和となる．このデータ転送時間は，2個のノードの割り当てられるモジュールの位置により決まる．また演算時間は，演算を表すノードの割り当てられるモジュールの処理時間により決まる．この演算で用いるモジュールの演算時間と，2個のモジュール間のデータ転送時間を以下のように定義する．

O_i(モジュール i の演算時間)

C_{ij}(モジュール i, j 間のデータ転送時間)

ここで，ロジックインメモリVLSIシステムは以下のような特徴を有するものとする．

- 粒度の小さい演算器，メモリ，レジスタなどのモジュールが空間的並列構造を持つ．
- システム全体の処理時間が最小となるようアロケーションされている．

本稿では，対象とするロジックインメモリVLSIシステムとして図2のようなメモリ，演算部間の相互結合網として，クロスバスイッチを持つものを考える．この演算部は1ステップで1個の演算のみ処理できるとし，その内部に1個のレジスタを持つとする．またメモリは，mワードの容量を持つとし，入出力はシングルポートとする．クロスバスイッチを用いて，モジュール間のデータ転送を並列に行うので，システム全体の処理時間は，全ての依存関係にあるノードの組の中で最大処理時間を持つものにより決まる．例えば図3においてはP3が最大処理時間になる．つまりこの最大処理時間を T_{max} とし，処理のステップ数を S とするとシステムの処理時間 T_{sys} は以下のようになる．

$$T_{sys} = T_{max} S \qquad (1)$$

S はスケジューリングが決定済みのデータフローグラフを用いるため，一定となるので，T_{max} を最小にすることで，システム全体の処理時間が最小になる．

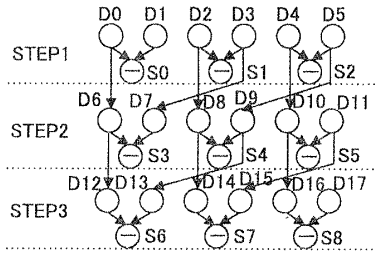

図 1: Sucheduled Data Flow Graph

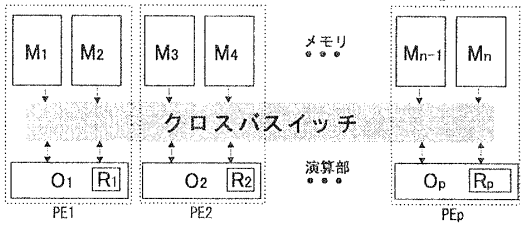

図 2: 対象とするアーキテクチャ

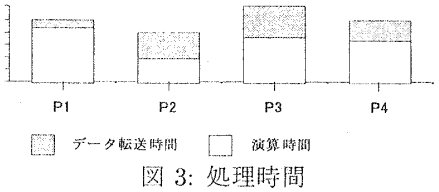

図 3: 処理時間

3．問題の定式化

このアロケーション問題を0-1整数計画問題として定式化する．変数として，x_{ij} を以下のように設定する．

High-Level Synthesis of a Logic-in-Memory VLSI system with an Interconnection Delay between Modules
Takashi HORII, Takahiro HANYU, Michitaka KAMEYAMA
Graduate School of Information Sciences, Tohoku University, Sendai-shi, 980-8579 Japan

$$x_{ij} = \begin{cases} 1 & (\text{ノード } N_i \text{がモジュール } j \text{に} \\ & \text{割り当てられる}) \\ 0 & (\text{その他}) \end{cases} \quad (2)$$

目的関数は，処理時間の最大値 T_{max} を考えればよい．ある依存関係にある2個のノード N_{i_1}，N_{i_2} があり，先にあるノードを N_{i_1}，後にあるノードを N_{i_2} としたときこの依存関係を $N_{i_1} \rightarrow N_{i_2}$ と表す．

まず $N_{i_1} \rightarrow N_{i_2}$ という関係にあるノード間に着目する．この処理時間は変数を用いて以下のように定式化される．

$$\sum_{j_1}\sum_{j_2}(C_{j_1j_2} + O_{j_2})x_{i_1j_1}x_{i_2j_2} \quad (3)$$

この値を，すべての依存関係にあるノード間について調べたときの最大値が T_{max} となる．

$$\max\Big(\sum_{j_1}\sum_{j_2}(C_{j_1j_2} + O_{j_2})x_{i_1j_1}x_{i_2j_2} \;\Big|\; \forall(N_{i_1} \rightarrow N_{i_2})\Big) \quad (4)$$

これが目的関数となり，これを最小にすることで，システム全体の処理時間 T_{sys} は最小となる．

次に制約条件を以下のように設定する．

条件 1 演算，記憶は1個のモジュールで実行される．
条件 2 演算器は1ステップで1個の演算が実行できる．
条件 3 メモリは同時に1個のデータがアクセスできる．
条件 4 メモリは m ワードのデータを記憶できる．
条件 5 レジスタは同時に1個のデータを記憶できる．

これを定式化するとそれぞれ以下のようになる．

$$\sum_j x_{ij} = 1 \quad (\forall i) \quad (5)$$

$$\sum_{i_o} x_{i_oj_p} \leq 1 \quad (\forall j_p) \quad (6)$$

$$\sum_{i_a} x_{i_aj_m} \leq 1 \quad (\forall j_m) \quad (7)$$

$$\sum_{i_s} x_{i_sj_m} \leq m \quad (\forall j_m) \quad (8)$$

$$\sum_{i_s} x_{i_sj_r} \leq 1 \quad (\forall j_r) \quad (9)$$

ここで i_o，i_s はそれぞれ同じステップにある演算，記憶のノード番号，i_a は同じステップにアクセスする記憶のノード番号，j_p，j_m，j_r はそれぞれ演算器，メモリ，レジスタのモジュール番号である．

式(5)から式(9)の制約下で，式(4)を最小にする変数の値を求めれば，ノードの割り当てられるモジュールが求まり，目的関数からシステムの処理時間が求まる．

4．応用例

この問題を SAD 演算に応用した例を示す．まず入力するデータフローグラフは図1とする．割り当てるアーキテクチャを図2とし，このモジュール間の転送時間 C_{ij} はシミュレーションから求め，表1に示す．また各演算部の演算時間は一定とする．これらを考慮し，0-1 整数計画問題として解いた結果，ノードの割り当ては図4のようになる．この結果から，図1中の $N_{D5} \rightarrow N_{D9}$ という依存関係のノードに着目すると，それぞれ D5 が M4，D9 が R0 に割り当てられている．この転送の種類，異なる PE 間でのメモリ→レジスタ転送が，表1より転送時間3となっていることがわかる．他の依存関係の全てについて調べると，この転送と PE1 から PE2 へのレジスタ→レジスタ転送以外は，全て同一 PE 内での転送となっており，この転送時間は3より小さく，最大転送時間は T_{max} は3となる．PE 間を転送しないアロケーションはないので，最小化が達成されていることがわかる．

5．まとめ

データ転送時間を考慮し，システムの処理時間を最小化するアロケーション法を示した．本方法を用いることにより，データ転送時間の短い，物理的に近接したモジュール間でデータ転送を行うロジックインメモリ VLSI システムを実現できる．今後は，モジュールを冗長に配置して，データ転送時間をさらに短縮する，ロジックインメモリ VLSI システムのアロケーション法について検討することが重要である．

参考文献

[1] B.Kim et al., "Parallel VLSI Processors for Robotics Using Multiple Bus Interconnection Networks," IEICE Trans. Fundamentals,E75-A,6,712/719,June,1992.

表 1: モジュール間転送時間

転送の種類	同一 PE 内	異なる PE 間
メモリ→演算器	2	3〜6
メモリ→レジスタ	2	3〜6
レジスタ→演算器	1	3
レジスタ→レジスタ	-	3,5

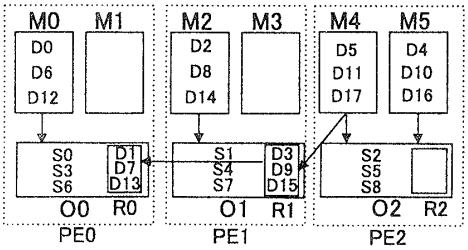

図 4: アロケーション結果