

非同期式プロセッサの性能評価の試み

5H-3

上野洋一郎

近 成人

南谷 崇

東京工業大学 工学部

1 はじめに

近年スイッチング遅延が数 psec の論理素子を実現されたという報告があるが、このような高速な素子を用いて同期式のプロセッサを設計する場合、配線遅延の問題が無視できない。その理由は、プロセッサの全体で同期をとるクロック信号が配線遅延の影響を受けて、素子の高速性を生かしたままでは同期信号として使えなくなるためである。

素子の高速性を充分利用する方法の一つは、クロック信号を用いない非同期式プロセッサを実現することである。しかしながら、現在まで非同期式のプロセッサの製作事例はほとんど存在しない。また、非同期式プロセッサの性能はアーキテクチャ、制御方式、回路構成などに大きく依存する。

そこで、我々は非同期式プロセッサに必要とされる制御方式や機能モジュールを明確にするために、非同期式プロセッサを Muller の C 素子などを用いて設計を行ない、どの要素がどの程度性能に影響を及ぼすかを評価した。

本稿では、この非同期式プロセッサの性能評価の方法を紹介するとともに、その結果を報告する。

2 プロセッサの構成

プロセッサの性能に寄与する要因に関して、同期式と非同期式の相違点を明確にすることを目的として、通常の同期式のアーキテクチャをそのまま非同期式プロセッサの目標アーキテクチャとして設計した。

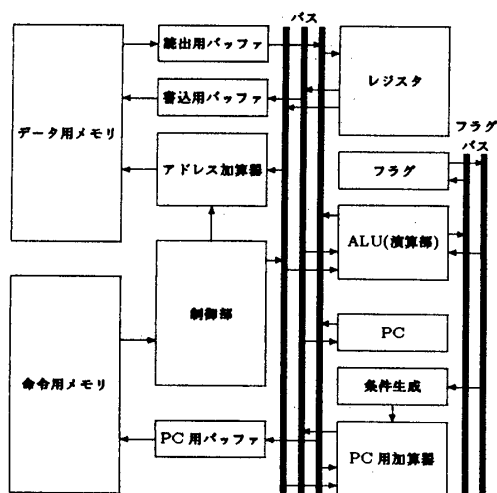


図1: プロセッサの構成

設計された非同期式プロセッサの諸元は以下の通りである。

- レジスタ間データ転送は2線2相式。
- アドレス・命令・データ語長 16bit。
- 16bit レジスタが8本。(1本は常に0を保持するレジスタ。)
- フラグ 4bit。(キャリー、ゼロ、オーバーフロー、ネガティブ)
- 1命令1語としてデコードを容易に。
- RISC型命令セット
- バス 16bit × 3。フラグバス 4bit × 2

- マスタースレイブ形レジスタにより、読み出し・書き込みを同時実行。

3 性能評価の方法

まだチップとして存在しない本プロセッサの性能評価は以下ののような手順で命令処理時間の期待値を計算することで行なった。

1. 一般的なプログラムにおける命令毎の使用頻度を調べる
 2. チップの簡易なフロアプランを行なう
 3. 命令毎の処理における path の配線長と通過するゲート段数からその path の所要時間を計算し、最長 path を求める
 4. 命令毎の最長 path に命令の使用頻度を乗じた値の合計をとる
- 所要時間の計算には以下の式を用いた。

$$\text{所要時間} = \frac{6\lambda}{\alpha c} (\text{path の配線長} + \text{ゲート段数} \times \beta)$$

λ チップの設計ルール

α チップ上での信号の伝搬速度の光速に対する比率

β 単位長(6 λ)当たりの信号の伝搬遅延に対するゲートの遅延の比率

c 光速

最終的な性能評価には命令フェッチ、ストア、ロード等のメモリアクセスも考慮すべきであるが、これらの実行ではメモリのアクセスタイムが性能の上限となる。メモリのアクセスタイムを見かけ上高速化する手法はあるがこれは非同期式に限ったものではない。そのため本稿ではメモリアクセスの評価を省く。

4 命令使用頻度の推定法

命令の使用頻度を正確に推定するには与えられた命令セットを生成するコンパイラとその命令セットを実行できるシミュレータが必要である。(シミュレータが必要な理由は、プログラム実行中の命令の出現頻度は処理のループや分岐によってコンパイラの生成したコードを単純に統計処理したものとは異なるからである。)しかしこれは大変な作業を必要とする。そこで、本プロセッサの命令セットに似た命令セットを持つプロセッサにおいて命令の使用頻度を調査して、命令に対応関係がとれるものはそのまま、とれないものは対応するように複数の命令に分解して使用頻度を推定した。その結果を表1に示す。

命令の分類	出現頻度
ロード	19.0%
ストア	12.6%
算術演算(レジスタ間)	15.7%
算術演算(レジスタ即値間)	11.7%
論理演算(レジスタ間)	7.3%
論理演算(レジスタ即値間)	5.1%
単項演算(シフト等)	3.6%
即値ロード	2.5%
相対ジャンプ	1.4%
レジスタジャンプ	2.1%
相対コール	1.7%
レジスタコール	0.0%
条件分岐	16.6%

表1: 命令の出現頻度

5 フロアプラン

5.1 回路の面積評価

与えられた回路図をチップに実現した場合に必要な面積を求めるのは容易ではない。同期型の回路ならば既存のブロックに分割することで面積を求めることもできるが、非同期型の回路ではそのような配置の実例も存在しない。

そこで、回路のおおまかな配置と配線を決定した後はその回路に要するであろう面積を以下の式を用いて回路に使われているゲートから推定した。(但しC素子はSR-FFにAND,ORをつないで実現したものと仮定した。)

$$\text{面積} = \sum (2 + \text{ゲートの入力線数})$$

そして、ある面積を持つ回路の外形は正方形になると考え、先に決定したレイアウトの中に配置した。

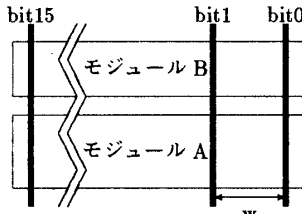


図2: バスの配置

また、バスは図2に示すようにモジュールの中を貫く形で構成する。この構成を行なった場合、バスに貫かれるモジュール同士ではバスのビット間の幅 w は揃えることが望ましい。そこで、 w はモジュールの中でもっとも幅の広いモジュールに揃えた。

5.2 簡易フロアプラン

フロアプランを決定する指標となるのは先に調べた命令使用頻度である。命令の使用頻度から更にモジュール間の情報の往來の頻度が得られるので、その情報を元にモジュールを配置する。

ただし、下位からのキャリーの有る加減算を行なう場合は特別な考慮が必要である。これは、図3,4を見るとわかるように、キャリーの情報は必ず加算器を横に往復してフラグに戻るわけだが、図3の場合はフラグを出力する前と戻ってきた後にも加算器の幅を往復することになる。それに対して図4はそのような情報の移動が無い。このように path の配線長が長くなるのが明白な配置は除外する。

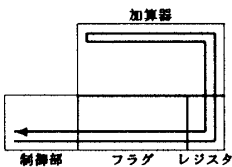


図3: 悪い配置の例

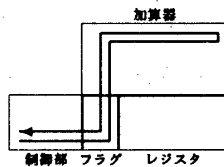


図4: 良い配置の例

以上の事実を考慮して行なったフロアプランの結果が図5である。

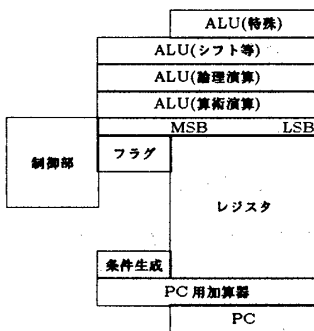


図5: フロアプラン

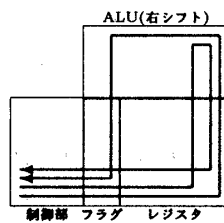


図6: 最長 path の例

6 命令毎の最長 path の決定

データの送り出し側であるレジスタではビット毎の回路は同一なのでゲート段数には差が生じないため、基本的にはバスの中で制御部からもっとも遠いビットの配線が最長 path となる。

例外として図6に示す右シフト等は制御部からもっとも遠いビットの path がALUへの行きと帰りで異なるので、各々の path の配

線長とゲート段数を評価してより長いものを選ぶ。また、モジュール内で複数の path が分岐合流をする場合も、各々の path を評価してより長いものを選択する。

7 性能評価

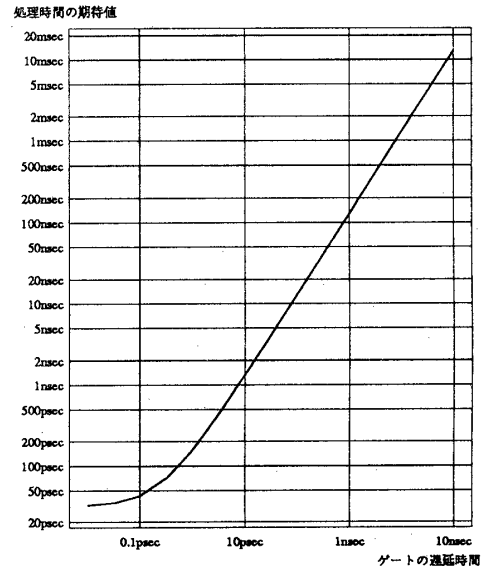


図7: 非同期式プロセッサの性能評価

最終的に得られた性能評価の結果を図7に示す。このグラフは、 λ を 0.1μ 、 α を光速の三分の一として β を変化させたものである。横軸がゲート一段の遅延時間を、縦軸が命令の処理(フェッチとデコードは除く)に要した時間の期待値である(3節の4で述べたように、非同期式プロセッサの性能は各命令の最長 path とその命令の出現確率の積の総和で得られる期待値である)。このグラフで右上がりの直線部分はゲート遅延が性能を決定する要素として支配的な領域で、水平部分は配線遅延が支配的な領域である。

従って、非同期式プロセッサは最長 path の期待値を下げるのが課題となる。最長 path の期待値を下げる方法は、出現頻度と最長 path の積が大きい命令から順に最長 path を短くすればよい。この結果として、出現頻度の低い命令の最長 path が大きくなったとしても期待値が下がっていれば性能は向上したことになる。

本プロセッサで最長 path の期待値を大きくしている主要因として、2線2相式データ転送、リップルキャリー加算器、マスタースレイブ形レジスタがあげられる。2線2相式データ転送は符号語状態とスペース状態を交互に繰り返すので、本プロセッサのアーキテクチャでは処理時間の半分は浪費される。回路設計が容易であったため採用したリップルキャリー加算器は算術演算の出現頻度が高いため、性能への影響が大きい。また、マスタースレイブ形レジスタの内容更新のオーバーヘッドはほとんどすべての命令で行なわれるので、これも性能への影響が大きい。

8 まとめ

本稿では非同期式プロセッサの性能評価として我々が設計したプロセッサを題材としてとりあげ、大まかではあるがそのゲート遅延と性能の関係を明らかにした。今後は性能評価で得られた事実から、より非同期式プロセッサに適したアーキテクチャや機能モジュール、回路構成を研究する予定である。

なお、本研究の一部は文部省科学研究費補助金「02452156」によって行なわれたものである。

参考文献

- [1] 南谷 崇. 同期式プロセッサの限界と非同期式プロセッサの課題. 信学技報, FTS90-45, December 1990.