

3P-5 多倍長演算機能を持ったプロセッサの実現方法の検討

*1 月江 伸弘

*2 高野 鉄平

*1*2 松永 俊雄

*1 東京工科大学大学院工学研究科

*2 東京工科大学工学部情報科

1 はじめに

計算機上で使用する変数の精度は、プロセッサに依存するため、安易に変数の精度を上げることができない。この場合ソフトウェアによる多倍長アルゴリズム [1] に頼ることになるが、現在市販されている計算機の演算速度を用いても膨大な実行時間を必要とすることが多い。

そこで、本検討ではアプリケーションに適した専用の LSI 設計が可能な FPGA を使用し、この多倍長アルゴリズムをプロセッサ上に実装することにより、多倍長精度を確保しつつ、演算の高速化を行うことをねらいとする。

本報告では、多倍長演算（加算，減算）のプロセッサ上における実行過程を述べ、その演算速度をソフトウェアで実施した場合と比較評価を行うことにする。

2 アーキテクチャ

2.1 基本構成

本プロセッサは、基本的なプロセッサ構造をできるだけ崩さず、多倍長演算を可能とさせることを目指して設計した。プロセッサの基本構成は、文献 [2] を参考にし、ALU 及びメモリ幅を 32bit とした 4 段パイプラインで設計を行った。ステージの構成を図 1 に示す。

多倍長命令は、32 ビット以上のデータを扱うことから、RR ステージと EX ステージとの間をループする形をとっている。

2.2 データフォーマット

多倍長データのフォーマットを図 2 に示す。

仮数部はユーザーが任意に指定した N により、 $N \times 32$ ビットのビット幅をもつことができる。ここで

An implementation method of a processor with a multiple precision arithmetic feature

*1 Nobuhiro Tsukie, *2 Teppei Takano, *1*2 Toshio Matsunaga

*1 Graduate School of Engineering, Tokyo University of Technology

*2 Dept. of Information Technology, School of Engineering, Tokyo University of Technology

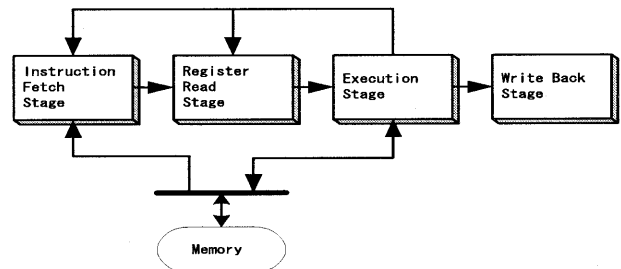


図 1: ステージの流れ

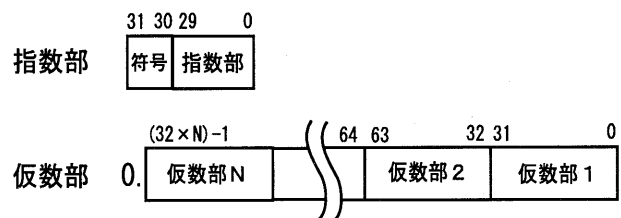


図 2: データフォーマット

N は 1 以上の整数である。指数部は 31 ビットとし、ゲタばき表現をとるものとしている。指数部の先頭には仮数部の符号を表す符号ビットを用意し、指数部と合わせて、32 ビットとしている。

これらのデータのメモリへの格納順番は、アドレスの小さい順から仮数部 1, 仮数部 2, ..., 仮数部 N , 指数部 + 符号ビットとしている。

2.3 多倍長命令

本プロセッサにおける命令長は、すべて 32bit の RISC 型で、3 オペランド形式を採用している。多倍長用加減算命令を表 1 に示す。

多倍長演算命令では、オペランドの指定に数値データがあらかじめ格納されているレジスタを指定するのではなく、多倍長データが格納されている最下位アドレスを格納したレジスタを指定する。rs, rt, rd のアドレス値は、初期の RR ステージ実行時において内部専用レジスタに格納され、その値をもとに処理される。

表 1: 多倍長用加減算命令		
命令	形式	意味
MADD	rs, rt, rd	$M[rd] \leftarrow M[rs] + M[rt]$
MSUB	rs, rt, rd	$M[rd] \leftarrow M[rs] - M[rt]$

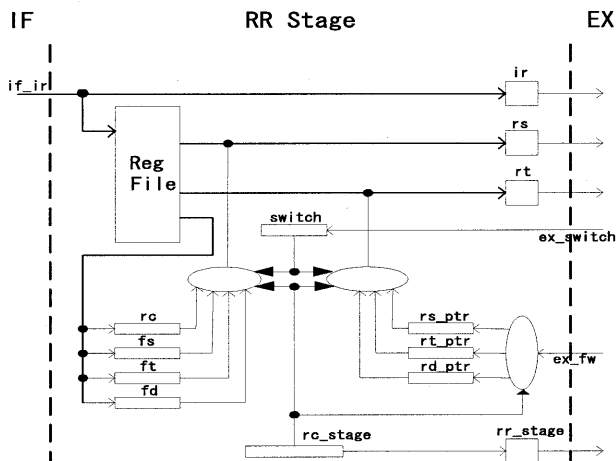


図 3: RR ステージのデータパス

3 データフロー

RR ステージ, EX ステージのデータパスを図 3, 図 4 に, 各図における記号の意味を図 5 に示す. 図 3, 図 4 には, 多倍長演算命令におけるデータパスのみ記述しており, 一般的な命令のデータパスは省いている. 本プロセッサは複数の専用レジスタを RR ステージ, EX ステージに配置することで多倍長演算を実現している.

RR ステージでは, 内部専用レジスタの値 (ループ回数, 演算対象が格納されているアドレス値) を演算工程によって, EX ステージへ送るものを切り替えている. EX ステージでは, RR ステージより送られてきた値, 及び EX ステージにある内部専用レジスタの値を使用し, 演算工程によって格納先アドレスの算出, メモリからデータの取り出し, 取り出したデータによる演算, 及び, 結果の保存を行う.

4 評価

論理合成は, ALTERA 社のセルライブラリを使用し, PARSHENON[3] の SFL コードからネットリストを作成, Quartus により MIPS 値を算出した. 想定デバイスは, APEX 社 EP20K100 としている. その結果, 動作周波数 20MHz で動作を確認している.

この結果から, 仮数部を $N = 10(2^{320} \div \text{有効桁数 } 100)$ とし, 10000 回の加算, 減算結果を, それぞれソフ

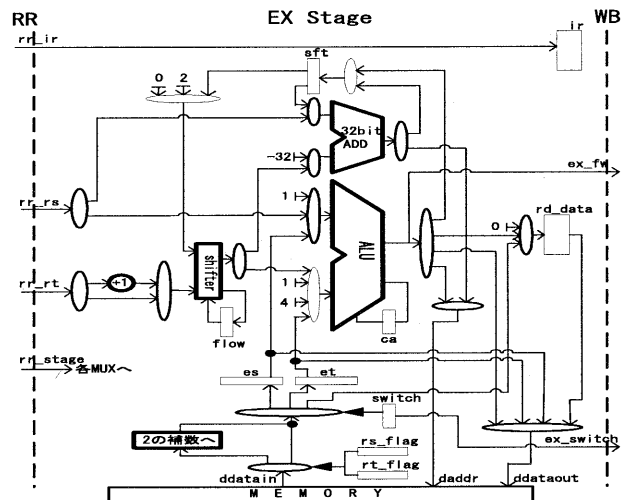


図 4: EX ステージのデータパス

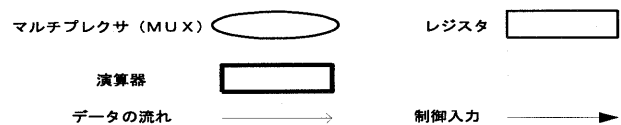


図 5: 記号の意味

トウェア上による多倍長計算と比較した (表 2). ソフトウェア上での計算に使用したプロセッサは Pentium II 266MHz, OS は FreeBSD4.0 を使用した.

表 2: 多倍長加減算命令の実測値 ($N = 10$)

演算工程	実測値 (HD)	実行時間 (SF)
多倍長加算	62.5 [ms]	1.45 [s]
多倍長減算	76.0 [ms]	1.45 [s]

5 おわりに

ソフトウェア上で実行される多倍長アルゴリズムをハードウェア化することで, 加減算処理を高速化することができた.

今後は, 現行の多倍長演算プロセッサの改良と乗除算演算の追加を行っていく予定である.

参考文献

- [1] D.E.Knuth(中川 圭介訳), 準数値算法 / 算術演算, サイエンス社, (1986).
- [2] David A.Patterson, John L.Hennessy (成田 光彰訳), コンピュータの構成と設計 第 2 版 上下巻, 日経 BP 社, (1999)
- [3] パルテノン研究会, 第 7 回パルテノン講習会テキスト, パルテノン研究会, (1999)