

VLSIソータ第2版の設計

2E-3

林 隆史<sup>†</sup> 喜連川 優<sup>††</sup> 伏見 信也<sup>†</sup> 田中 英彦<sup>†</sup> 元岡 達<sup>†</sup>

(<sup>†</sup>東京大学工学部 <sup>††</sup>東京大学生産技術研究所)

1. はじめに

現在、ハードウェアソータは種々のデータ処理、殊に関係代数処理に於いて、その処理時間を大幅に短縮させるものとして期待されている。我々はこれまでN個のレコードを  $O(\log_2 N)$  個のモジュールを用いて  $O(N)$  時間でソータを行うパイプラインマージソータのLSI化を進めてきたが、パイプラインマージソータ、及び他に提案されているハードウェアソータは実際の使用環境に於いて種々のパラメータ(レコード長、ストリーム長等)の変動に対するソータの柔軟性が乏しく、その高速性に対する大きな代償となっていた。

今回は、これらパラメータの変動に対して柔軟に対処するための拡張機能を実現すべく設計を進めてきたVLSIソータ第2版について報告する。

2. VLSIソータ第2版の機能

パイプラインマージソータは2-wayマージソータを行うプロセッサを1次元状にパイプライン結合した構成をとるが、従来の第1版ではMTM(Merge Top Memory)を設けて、マージペアの片方のレコードをMTMに格納し、メモリアクセス回数を減らして高スループットを得る設計となっている。これに対して第2版では、MTMを設けないことによって、回路規模の縮小と制御機構の簡単化を行った後、更に次のような拡張機能を実装する。

(1) マルチキーソータ

レコードの先頭に存在する複数のキーに対して昇順、及び降順ソータを任意に設定できる。

A	D	A	D	Non-Key
---	---	---	---	---------

A: 昇順 D: 降順

Fig. 1 マルチキーソータに於ける昇順/降順指定

(2) Length Tuning

入力レコード長Xが設計レコード長Lと異なる場合、各プロセッサのメモリには空き領域が生じる。X>Lの場合この空き領域をlength tuningにより利用し、より多くのレコードをソータ可能とする。

(3) ストリームバイパス機構

ソータのパワーをM(レコード)、入力ストリーム長を

Y(レコード)とする。Y<Mの場合、ダミーレコードを付加してY=Mとしてソータを行うことも可能であるが、ソータ終了を検出して不要なプロセッサをバイパスすることで高速化が可能である。

(4) Stream Driven Connection

Y>Mの場合には、入力ストリームがソータ1台のパワーを越えているためstream driven connectionによりソータを複数台結合してソータを行う。

これらの拡張機能を実現するために次のようなフラグを用いて制御を行う。

- dr...dummy record
- sos...start of string
- ls...last string
- sbp...stream bypass
- nmp...no merge pair
- eos...end of stream

3. Length Tuning アルゴリズム

i番目のプロセッサPiは $2^i \cdot L$ のメモリと $2^{i+1} \cdot L$ の論理空間を有している(2)。この $2^{i+1} \cdot L$ の論理空間にレコードをできるだけ詰め込みメモリの使用効率を最適化することをlength tuningと言う。このときの出力ストリング長Ni(レコード)は $N_i = \lfloor 2^{i+1} \cdot L / X \rfloor$ より求まる。この長さNiのストリングをPiが出力するためには、長さ $\lfloor N_i / 2 \rfloor$ と $\lceil N_i / 2 \rceil$ のストリングをマージすればよい。更に $\lfloor N_i / 2 \rfloor$ と $\lceil N_i / 2 \rceil$ を2分割してPi-1の入力ストリング長が求まる。この分割をk回繰り返して $\lceil N_{i-k} / 2 \rceil \leq 1$ となるkを求める。そして、 $j < i-k$ なるPjはストリームを素通しし、Pi-kが長さ1 or 2のストリングを出力することによりPiは長さNiのストリングをその論理空間へ詰め込むことができる。

N6=11でlength tuningを行う場合をFig. 2に示す。この時k=3より、P1, P2はストリームをバイパスし、P3よりマージを開始することになる。

以上のlength tuningの制御を行うには、Piに対して入力ストリングがマージペアを持つかどうかを指示してやる必要がある。ここではnmp flagをマージペアを持たないレコードに付加して、Piを素通しさせることにした。また、ストリーム内でストリング長が変動することがあるためストリングが出力されない時がある。この時にはdummy recordを出力して、パイプラインに乱れが生じないよ

うにする。dummy record には dr flagが付加される。

Fig. 2 の場合の nmp, dr フラグ操作, 及びタイミングチャートをFig. 3 に示す。

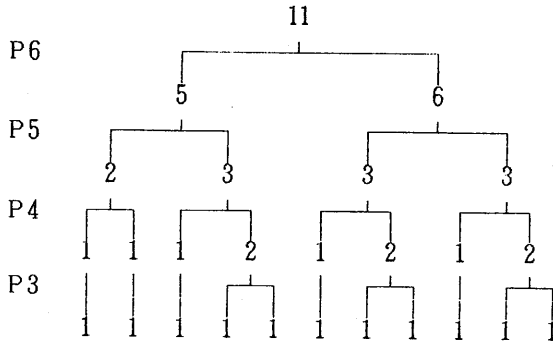


Fig. 2 Length Tuning Algorithm

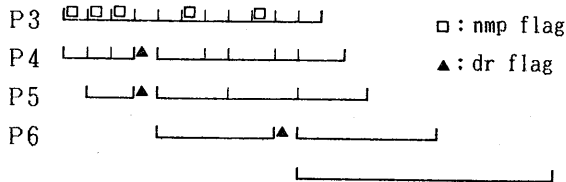


Fig. 3 nmp, dr Flag Manipulation for Length Tuning

4. ストリームバイパス機構

ストリームバイパス機構を実現するためには、ソート終了の検出が重要となる。この検出は ls, eos 2つのフラグを用いて、次のようにして行う。eos は入力ストリームの最後に、ls は入力ストリームの最後のストリングの先頭に付加されている。各プロセッサは上記と同様にフラグを付加してストリームを出力する。この時、ソートが完了したストリームに於いては、ls が先頭に、eos が最後に付加されており、入力ストリームの先頭で ls を検出したプ

ロセッサはストリームを素通しすればよい。

5. Stream Driven Connection

Stream driven connection は複数個のソータを結合して更に大容量のソータを構成する結合方式である〔1〕。本方式では、1次元状に接続された各ソータは内部各プロセッサのメモリが一杯になるM-1個のレコードを入力した時点でマージを休止し、それ以後のレコードをバイパスして次ソータへ送る。そして最終ストリングが入力された時点で、最終ストリングをM個目のレコードと考えてマージを再開する。本方式は ls, sbp 2つのフラグによって制御され、各ソータの入力ストリームのM個目のレコードに sbp が、最終ストリングに ls が付加される。各ソータの各プロセッサはsbp を検出するとそれ以後のレコードをバイパスし、lsを検出した時点からマージを再開する。

長さ11の入力ストリングを Stream driven connection されたパワー8のソータ2台によってソートを行う場合のフラグ操作, 及びタイミングチャートをFig. 4 に示す。

Fig. 4 に示されるように、本方式ではソータレベルに於いてもパイプライン処理が可能である。

6. おわりに

VLSIソータ第2版に於ける拡張制御方式について考察した。現在、プロセッサのレジスタトランスフェレベルに於ける設計・記述を行い、シミュレーションによって制御の正当性を確認している。

《 参考文献 》

- 〔1〕 林他：パイプラインマージソータのLSI化 情報処理学会第26回全国大会 4F-4, 1983
- 〔2〕 喜連川他：パイプラインマージソータの構成 信学技報 EC82-32, 1982

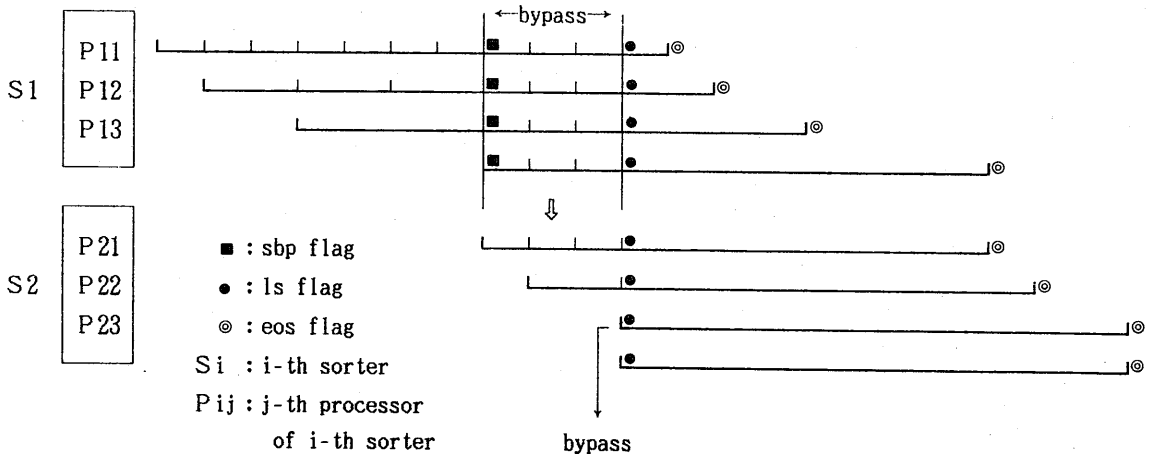


Fig. 4 sbp, ls Flag Manipulation for Stream Driven Sorter