

パイプラインマージソータの構成

4P-4

＝ LSI 化に関する一検討 ＝

林 隆史, 喜連川 優, 伏見 信也, 田中 英彦, 元岡 達

(東京大学 工学部)

1. はじめに

データの高速ソータリングは種々の分野で必要とされる基本的な操作である。我々はデータベースマシンGRACEの基本要素としてデータストリームに沿った処理を可能とするハードウェアO(N)ソータを開発してきた[1, 2]。すでにTTLによる試作を終え、今回そのLSI化についてを検討したので報告する。

2. ソータの構成

ソータアルゴリズムはパイプライン化されたマージソータを基調としている。今、 $N (=K^n)$ 件のレコードのソータを行なうものとする、 K -way mergeを行なうプロセッサを $n (= \log_K N)$ 台用意し、これらを一次元状に結合する。第 i 番目のプロセッサは第 $i-1$ 番目のプロセッサから送られてくる K^{i-1} 件のレコードからなるソータされたストリングを K 本マージして K^i レコードからなる一本のストリングを生成し、第 $i+1$ 番目のプロセッサへ送る。

図1にプロセッサの入出力ピン、図2にプロセッサの内部構成を示した。

3. HSLによる記述

HSLは電々公社で開発された階層仕様記述言語である[3]。その特徴として以下のような点がある。

- (1) 論理LSIの階層化設計を各階層に対してサポートする。
- (2) コンパイラ、マクロ展開、切り出し処理などの機能がHIDEMAPシステムとして用意されており、さらにシミュレーションを行ない動作確認をすることが可能である。

このHSLを用いてソータを記述するにあたり、我々はソータを以下のように5つの機能ブロックに分割した。(図2)

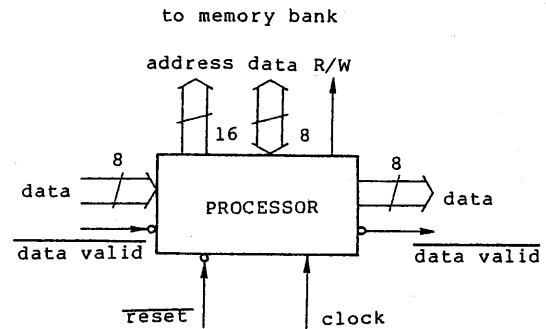


図1. プロセッサの入出力ピン

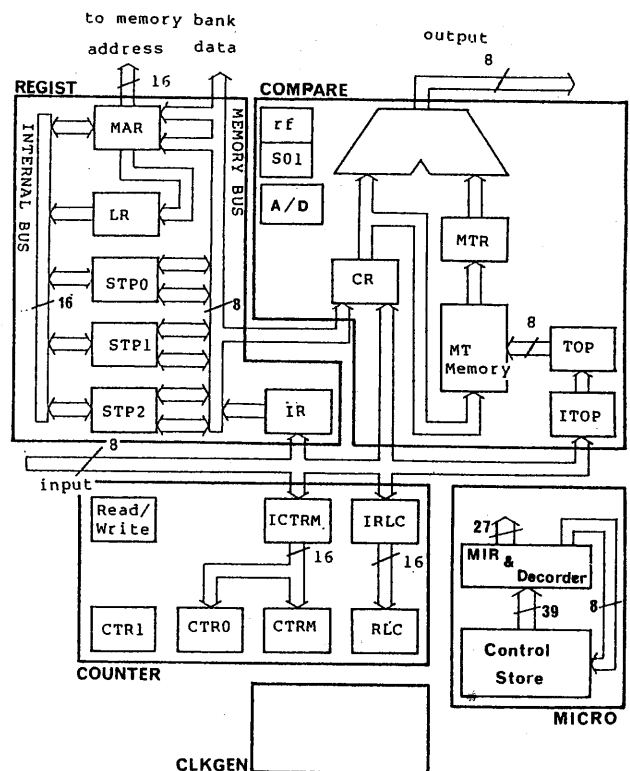


図2. プロセッサの内部構成及び機能ブロック分割

- (1) REGIST : MAR、LRなどlinked list用レジスタ群及びそれら进行操作するためのメモリバス、内部バス。
- (2) COMPARE : CR、MTRなどデータ比較制御回路。
- (3) COUNTER : レコード長、ストリング長の管理用レジスタ群。
- (4) MICRO : マイクロ命令レジスタ、コントロールストア及びデコーダ。
- (5) CLKGEN : クロックジェネレータ及びデコーダ。

HSLの記述例として、ソータで用いられている16bit up/down counterのHSL記述の一部を図3に、その階層構造を図4.1及び図4.2に示した。

尚、このソータの回路規模としてはコントロールストアを除いて4000ゲート程度であるが、HSLの階層性及びライブラリを効果的に利用することにより、ソータ全体の記述は1800行程度に収めることができた。

また、当ソータは既にTTLによる試作を終了していたため、HSLによる記述は比較的容易であり約1.5人月で終了した。

4. おわりに

このソータのLSI化に関しては今後詳細なシミュレーションを行なった後、ルータ(自動配置プログラム)を用いてマスクパターンを作成していく予定である。その他、拡張された機能のLSI化についても検討をすすめている。

5. 謝辞

今回の研究は電々公社武蔵野電気通信研究所の御協力により実現したものであり、集積応用研究室の方々に感謝致します。

<< 参考文献 >>

- [1] 喜連川、伏見、桑原、田中、元岡
「パイプラインマージソータの構成」
信学技報 EC 82-32 1982
- [2] 桑原、喜連川、伏見、田中、元岡
「GRACEに於けるソーティングユニットの構成」
情報処理学会第24回全国大会 4G-5 1982
- [3] 「階層仕様記述言語 HSL」
日本通信技術株式会社

```

NAME      :UDCNTR;
PURPOSE   :LOGSIM;
LEVEL     :MODULE;
EXT       :D<0:15>,.Q<0:15>,.CLK,.LD,.EN,.UD;
INPUTS    :.D<0:15>,.CLK,.LD,.EN,.UD;
OUTPUTS   :.Q<0:15>;
TYPES    :UDCTR4,DR5,IV010;
UDCTR4   :UDC<0:3>;
OR5      :OR<0:2>;
IV010    :IV;
#####
DLN<0:3>  =FROM(.D<0:3>)          TO(UDC<0>.D<0:3>);
DLN<4:7>  =FROM(.D<4:7>)          TO(UDC<1>.D<0:3>);
DLN<8:11> =FROM(.D<8:11>)         TO(UDC<2>.D<0:3>);
DLN<12:15>=FROM(.D<12:15>)        TO(UDC<3>.D<0:3>);
QLN<0:3>  =FROM(UDC<0>.Q<0:3>)    TO(.Q<0:3>);
QLN<4:7>  =FROM(UDC<1>.Q<0:3>)    TO(.Q<4:7>);
QLN<8:11> =FROM(UDC<2>.Q<0:3>)    TO(.Q<8:11>);
QLN<12:15>=FROM(UDC<3>.Q<0:3>)    TO(.Q<12:15>);
CILN<0:3> =FROM(UDC<0>.C<0:3>)    TO(OR<0>.1:4);
C2LN<0:3> =FROM(UDC<1>.C<0:3>)    TO(OR<1>.1:4);
C3LN<0:3> =FROM(UDC<2>.C<0:3>)    TO(OR<2>.1:4);
INCLN<0>  =FROM(.EN)              TO(OR<0>.5,UDC<0>.E);
INCLN<1:2>=FROM(OR<0>.1:6)        TO(OR<1>.2,5,UDC<1>.E);
INCLN<3>  =FROM(OR<2>.6)          TO(UDC<3>.E);
CLKIN     =FROM(.CLK)              TO(IV.1);
CLKLN     =FROM(IV.2)              TO(UDC<0>.CLK,UDC<1>.CLK,
UDC<2>.CLK,UDC<3>.CLK);
LDLN      =FROM(.LD)               TO(UDC<0>.LD,UDC<1>.LD,
UDC<2>.LD,UDC<3>.LD);
UPDNLN    =FROM(.UD)               TO(UDC<0:3>.UPDN);
END;
    
```

```

NAME      :UDCTR4;
PURPOSE   :LOGSIM;
LEVEL     :SMODULE;
EXT       :D<0:3>,.CLK,.LD,.E,.Q<0:3>,.C<0:3>,.UPDN;
INPUTS    :.D<0:3>,.CLK,.LD,.E,.UPDN;
OUTPUTS   :.Q<0:3>,.C<0:3>;
TYPES    :UDBCTR,UCCONT,IV010;
UDBCTR   :BC<0:3>;
UCCONT   :UDCC;
IV010    :N1,N2,N3;
#####
DLN<0:3>  =FROM(.D<0:3>)          TO(BC<0:3>.D);
YLN<0:3>  =FROM(UDCC.Y<0:3>)      TO(BC<0:3>.E);
QLN<0:3>  =FROM(BC<0:3>.Q)        TO(.Q<0:3>);
CLN<0:3>  =FROM(BC<0:3>.C)        TO(.C<0:3>);
CLN<0:2>  =FROM(BC<0:2>.C)        TO(UDCC.C<0:2>);
ELN       =FROM(.E)               TO(UDCC.E);
CLKIN     =FROM(.CLK)              TO(N1.1);
LDLN      =FROM(.LD)               TO(N2.1);
CLKLN     =FROM(N1.2)              TO(BC<0>.CLK,BC<1>.CLK,
BC<2>.CLK,BC<3>.CLK);
LDLN      =FROM(N2.2)              TO(BC<0>.LD,BC<1>.LD,
BC<2>.LD,BC<3>.LD);
UPDNIN    =FROM(.UPDN)             TO(N3.1);
UPDNLN    =FROM(N3.2)             TO(BC<0:3>.UPDN);
END;
    
```

図3. カウンタのHSL記述

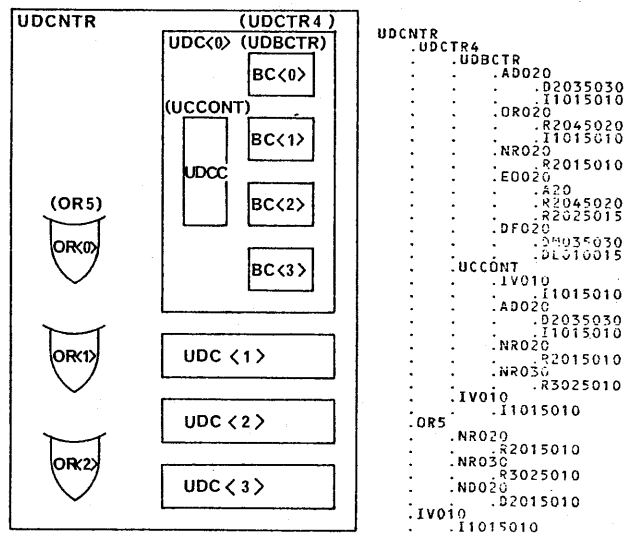


図4.1 図4.2

図4. カウンタの階層構造