

## 機能分散型デュアルプロセッサ構成による 高機能グラフィック・ディスプレイの設計と評価†

浜田長晴<sup>††</sup> 小林芳樹<sup>††</sup> 高藤政雄<sup>††</sup>

グラフィック・ディスプレイは、最近の LSI 技術の進展に支えられて性能対価格比を著しく向上させている。筆者らもマイクロプロセッサやランダムアクセスメモリなどの最新の LSI 技術をとりいれ、端末サイドでのインテリジェンスの強化を追求したカラー・グラフィック・ディスプレイを開発した。

本論文の目的は、下記の3点である。

- 1) グラフィック・ディスプレイの高機能化、特に端末における図形データの生成管理機能とその高速処理を実現するのに好適な機能分散型デュアルプロセッサシステムの性能評価手法を確立すること。
- 2) 上記性能評価手法を利用して表示処理性能や所要バッファ容量などの定量的評価に有効な設計チャートを作成すること。
- 3) この設計チャートの有効性を試作システムの実測値との対比で検証すると同時に、将来の性能改善のための見通しを得ること。

このため、グラフィック表示画面中の直線表示本数を用いて表示処理速度を推定するモデルを提案し、通常の8ビットマイクロプロセッサと図形発生専用プロセッサとの組み合わせで7,000本/秒の高速表示が実現できることを示した。

### 1. ま え が き

グラフィック・ディスプレイは1980年代のマン・マシンコミュニケーションの主役としてさらに発展することが期待されている<sup>1),2)</sup>。特に、最近のLSI技術の進歩を積極的にとりいれて性能対価格比を向上させようとする動きが顕著であり<sup>3)-5)</sup>、各社から新製品が次々に発表されている。

筆者らも、このような動向をいち早く察知し、端末としてのインテリジェント化を追求した高機能カラー・グラフィック・ディスプレイを開発した<sup>9),10)</sup>。

これらの最近のディスプレイに共通して言えることは Computing Power (Processing Power) 増強のためミニコンピュータやマイクロコンピュータの複合体になっていることである。文献5)~8)は、ミニコンピュータを中心にした大がかりなシステムであり、高度な図形処理機能を有しているが、システム全体の表示処理能力については評価されていない。文献3)はマイクロコンピュータを用いて、ホストからの図形コマンドを解釈させた場合の転送速度と表示速度との関係を求めているが端末で図形データの管理を行っていないので全体としての表示処理能力は推定されていない。

† System Design and Evaluation of Intelligent Graphic Display Terminal with Functionally Distributed Dual Processing Architecture by NAGAHARU HAMADA, YOSHIKI KOBAYASHI and MASAO TAKATOO (Hitachi Research Laboratory, Hitachi Ltd.).

†† (株)日立製作所日立研究所

い。

本論文では機能分散型のデュアルプロセッサシステムを適用したグラフィック・ディスプレイの表示処理性能を定量的に把握するため、表示画面内の直線本数から表示速度を推定するモデルを提案し、これを用いてプロセッサの処理速度やバッファ容量を求める設計チャートを作成している。さらに、その結果を応用して試作した装置にいくつかの表示画面を表示させて評価し、先のモデルが有効であることを確認している。また、本論文で提案している単純なデュアルプロセッサ構成でも両プロセッサの処理速度比を適切に選定すれば7,000本/秒の高速表示を実現し得ることを明らかにしている。

### 2. 機能分散型システムの概要と設計上の問題点

#### 2.1 システム構成

筆者らが最近開発した、機能分散型デュアルプロセッサ構成の高機能グラフィック・ディスプレイのシステム構成と表示の概念図を図1に示す。本ディスプレイは下記のような4個の主要ブロックからなっている。

- (1) コマンド・リスト・メモリ; 表示図形の原始データ(図形の種類, 座標, 倍率, etc.)を図形コマンドの集合として記憶しているメモリ。

- (2) メイン・プロセッサ(MP); ホスト計算機や

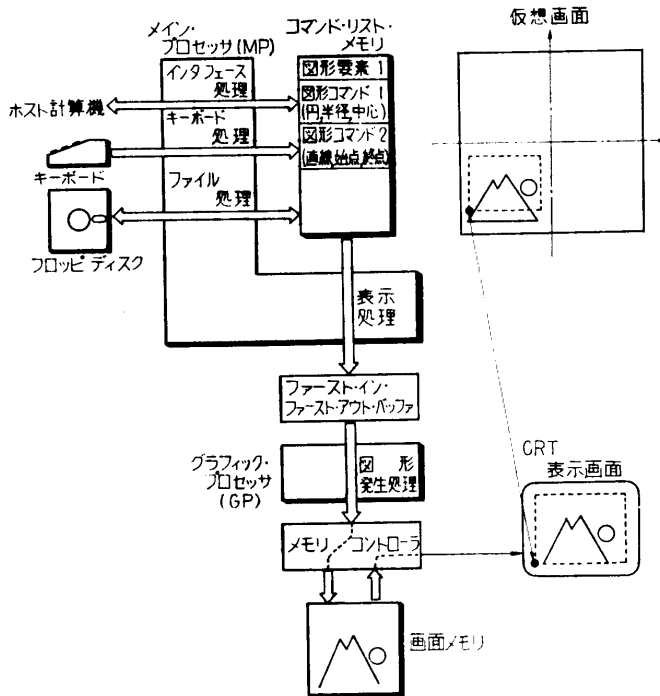


図1 グラフィック・ディスプレイのシステム概念図

Fig. 1 Conceptual block diagram of graphic display terminal.

フロッピー・ディスクなど外部機器との情報の授受を制御し、コマンド・リストの編集管理を司る汎用マイクロプロセッサ。

(3) グラフィック・プロセッサ(GP); 図形コマンドを解釈し、画面メモリに図形を生成・描画する専用プロセッサ。

(4) 画面メモリ; 表示画面に対応した絵素(ドット)単位の図形情報を記憶し、CRTをリフレッシュするためのメモリ。

本システムのグラフィック・ディスプレイとしての特長や機能はすでに報告されている<sup>9),10)</sup>ので省略する。

## 2.2 設計上の問題点

図1より明らかなように、本システムはMPとGPの2種類のプロセッサをファースト・イン・ファースト・アウト(FIFO)バッファを介して接続し、両者を機能分散型のデュアル・プロセッサとして並列動作させることによって表示処理性能の向上をはかろうとするものである。このようなシステムの設計においては次のような点を明らかにする必要がある。

(1) 両プロセッサにどのような機能を分担させるか。

(2) 両プロセッサにどのような処理性能を与える

か。

(3) 上記(1), (2)によって最終的な表示処理能力としてどの程度のものが得られるか。

これらを個々に議論することは困難であるが、(1)についてのマクロな結論は文献9)で報告したように、MPにコマンド・リストの管理を行わせ、GPに座標変換などの描画処理を行わせるのが最適であり、ここでは主として(2)および(3)について論じる。特に、両プロセッサの処理性能から必要なバッファ容量や、最終的な表示処理速度を推定するモデルを設定し、そのような値が容易に求められる図表を作成しておくことと便利である。

## 3. 表示処理性能評価モデル

本ディスプレイの表示処理性能は、いかに効率よくMPとGPとが並列動作し得るかということによって決まる。この動作形態は、両者が完全に機能分割されているため、FIFOバッファを介しての単一窓口サービスによる最も単純な待行列モデルとして評価できる。

まず、FIFOバッファに対する図形コマンドやデータの到着頻度のモデルを設定する。

コマンド・リスト・メモリの具体的な構成例は図2のようになっている。すなわち、コマンド・リスト・メモリは、リスト全体の管理情報を記憶しているエリア、共通な図形をサブルーチンとして記憶しているエリア、および拡大、回転、移動など図形操作の対象となる図形要素の集合として図形データを記憶しているエリアからなっている。さらに図形要素はいくつかの図形コマンドや図形サブルーチンの集合であり、図形コマンド自体も直線コマンドなどは複数本の線分の集合となっている。

このようなコマンド・リストを表示する場合、MPの一般的な表示処理フローは図3に示すように3重のループで処理される。すなわち、個々の図形コマンド内で必要なデータをとり出しFIFOにセットするループ1と、図形要素内の図形コマンドをサーチしGPが解釈できるフォーマットに変換してFIFOにセットするループ2と、全コマンド・リスト内の図形要素をサーチし拡大・回転・移動の有無に応じた付加情報をつけてFIFOにセットするループ3からなっている。したがって、1画面の表示処理のためには、各ループの

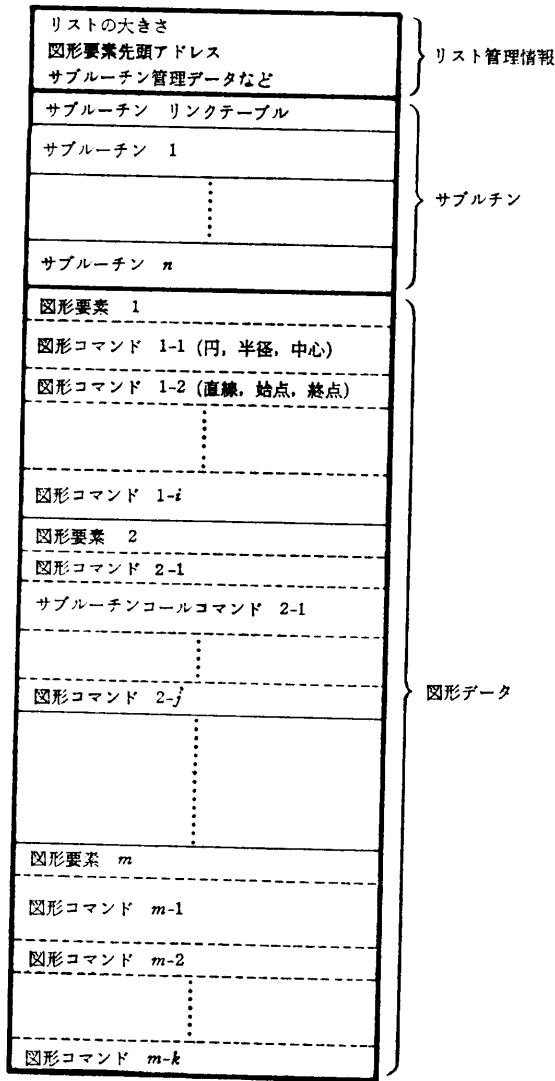


図 2 コマンド・リスト・メモリの構成例

Fig. 2 Contents example of command list memory.

処理時間やループ回数がリストの内容によって変化するため、FIFO バッファへの図形コマンドやデータなどの到着間隔はランダムに変わり、その平均到着間隔、 $\bar{t}_a$  は(1)式ようになる。

$$\bar{t}_a = \frac{t_{MP}}{l_1 \times l_2 \times l_3} \div \frac{t_{MP}}{l_1 \times l_2 \times l_3} = \bar{t}_1 + \bar{t}_2 \times \frac{1}{l_1} + \bar{t}_3 \times \frac{1}{l_1 \times l_2} \dots \dots \dots (1)$$

ただし、 $t_{MP}$  は表示処理に要した MP の全処理時間、 $l_1 \sim l_3$  はそれぞれのループのループ回数、 $t_1 \sim t_3$  はそれぞれのループの処理時間であり、簡単化するため  $l_1 \times l_2 \times l_3 = l_1 \times l_2 \times l_3$  とした。

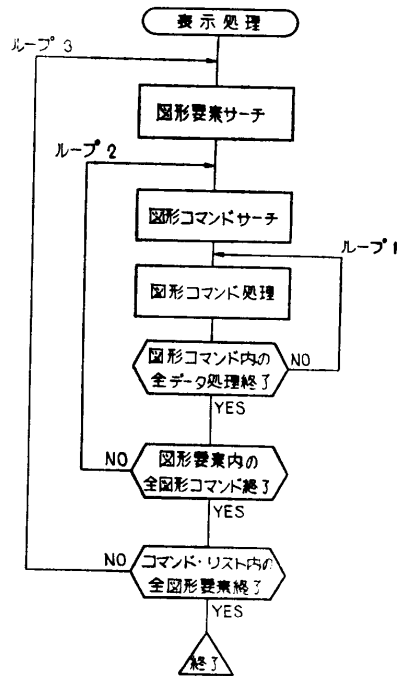


図 3 メイン・プロセッサの表示処理フロー

Fig. 3 Display data processing flow of main processor.

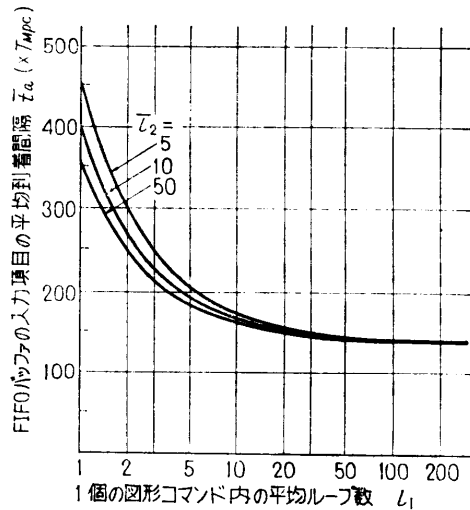


図 4 平均到着間隔の表示内容による変化

Fig. 4 Effects of display contents for average interarrival time.

$l_1 \sim l_3$  は表示画面の内容によって、 $\bar{t}_1 \sim \bar{t}_3$  は MP の処理能力によって変化する。しかし、 $\bar{t}_1 \sim \bar{t}_3$  は MP に通常の汎用マイクロプロセッサを使用するものとすれば、処理クロック数で正規化でき、 $\bar{t}_1 = 140 \times T_{MPC}$ 、 $\bar{t}_2 = 200 \times T_{MPC}$ 、 $\bar{t}_3 = 600 \times T_{MPC}$  ( $T_{MPC}$  は MP の処

理クロック・サイクル・タイム) 程度である。

したがって、平均到着間隔  $\bar{t}_a$  は次のようになる。

$$\bar{t}_a = \left( 140 + 200 \times \frac{1}{l_1} + 600 \times \frac{1}{l_1 \cdot l_2} \right) \times T_{MPC} \dots\dots\dots (2)$$

これを図示すると 図 4 のようになり、通常の画面では  $l_1$  は 2~3,  $l_2$  は 10~50 なので  $\bar{t}_a \approx 240 \times T_{MPC}$  と見てよい。

次に、GP の処理時間、すなわちサービス時間のモデルを設定する。GP の処理内容としては動作モードや座標修飾データなどを制御レジスタにセットするだけの単純な処理から座標変換・クリッピング・ドット展開といった高度な処理まであり、処理時間はランダムと言える。

図 5 に、最も一般的な直線の表示コマンドに対する GP の処理フローを示す。すなわち、表示色や線種(実線、点線、etc.)などの制御データをそれぞれのレジスタにセットする前処理部分と、直線を表す 2 点の座標データに対し、あらかじめ規定された座標変換を行い、その値が設定されたウインドウ内に入っているかどうかを判定するクリップ処理を経て絵素(ドット)位置を決定するドット展開処理を直線コマンドで規定された複数本の線分全てが終了するまで繰り返す後処理部分とからなっている。

したがって、GP の平均サービス時間  $\bar{S}$  は次のよう

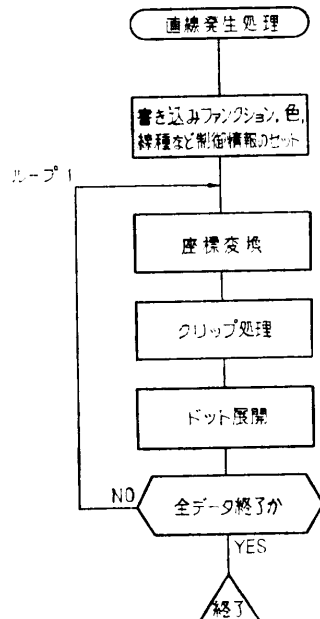


図 5 グラフィック・プロセッサの直線発生処理フロー  
Fig. 5 Vector generating flow of graphic processor.

になる。

$$\bar{S} = \frac{\sum_{i=1}^{n_{VEC}} T_{VEC_i} + \sum_{j=1}^{n_{\overline{VEC}}} T_{\overline{VEC}_j}}{n_{VEC} + n_{\overline{VEC}}} \doteq \frac{\sum_{i=1}^{n_{VEC}} T_{VEC_i}}{n_{VEC}} = \frac{k_1 + (k_2 + k_3 \bar{\pi}) l_1}{l_1} \times T_{GPC} \dots\dots\dots (3)$$

ただし、 $n_{VEC}$  は直線コマンドの数、 $n_{\overline{VEC}}$  は直線コマンド以外のコマンドの数、 $T_{VEC_i}$  は  $i$  番目の直線コマンドの処理時間、 $T_{\overline{VEC}_j}$  は  $j$  番目の非直線コマンドの処理時間、 $k_1 \sim k_3$  はそれぞれ直線コマンドの前処理、座標変換およびクリップ処理、および 1 ドットの展開処理に必要な処理サイクル数、 $\bar{\pi}$  は直線の平均ドット数、 $l_1$  は直線コマンド内の平均直線本数 (MP の  $l_1$  と同じ)、 $T_{GPC}$  は GP の 1 処理サイクル時間であり、通常の画面では  $n_{VEC} \gg n_{\overline{VEC}}$  となっている。

16 ビットの図形発生専用プロセッサの場合、 $k_3=3$  となり式(3)は次のようになる<sup>10)</sup>。

$$\bar{S} = \frac{60 + (k_2 + 3 \times \bar{\pi}) l_1}{l_1} \times T_{GPC} \dots\dots\dots (4)$$

ただし、拡大・回転・クリップなしの場合  $k_2=100$   
 拡大・回転・片端クリップの場合  $k_2=300$   
 拡大・回転・両端クリップの場合  $k_2=600$

式(4)を図示すると 図 6 になり、グラフィック・ディスプレイでは通常  $\bar{\pi}=50$  で評価するため  $\bar{S} \approx 450 \times T_{GPC}$  と見てよい。

通常の表示画面では、直線コマンドが表示コマンドの大部分であるため、ディスプレイの表示処理性能モデルの第一ステップとして直線表示処理性能が推定できれば十分である。したがって、このような処理性能モデルとしては、前述の平均到着間隔  $\bar{t}_a$  のランダム

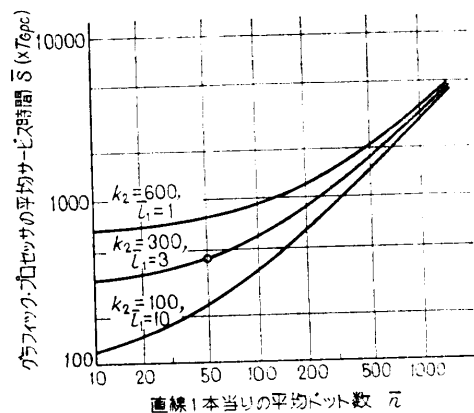


図 6 グラフィック・プロセッサの平均サービス時間  
Fig. 6 Average service time of graphic processor.

到着、平均サービス時間  $\bar{S}$  のランダムサービスを有する単一サービス窓口の待行列モデルの適用が可能である。

すなわち、FIFO バッファの中でサービスを待っている項目の数  $w$  の平均値  $\bar{w}$ 、およびシステム内の項目数  $q$  が  $N$  個以上になる確率  $P(q > N)$  はシステムの利用率を  $\rho$  として次式のように与えられる<sup>11)</sup>。

$$\bar{w} = \frac{\rho^2}{1-\rho} = \frac{\bar{S}^2}{\bar{t}_a(\bar{t}_a - \bar{S})} \quad \dots\dots\dots (5)$$

$$P(q > N) = 1 - \sum_{q=0}^N (1-\rho)\rho^q = \rho^{N+1} = \left(\frac{\bar{S}}{\bar{t}_a}\right)^{N+1} \quad \dots\dots\dots (6)$$

さらに、表示処理性能の最も有効な尺度となる全コマンド・リストの表示処理時間  $T_{DISP}$  を推定すると以下ようになる。

すなわち、到着間隔  $\bar{t}_a$  が、FIFO バッファ内の待項目数  $w$  が 0 になってしまう時間より短かければ平均サービス時間  $\bar{S}$  でサービスされ、長ければその到着間隔  $\bar{t}_a$  でサービスされることになり、表示処理時間  $T_{DISP}$  は次のようになる。

$$T_{DISP} = V_n \times \left\{ P(t_a \leq w \times \bar{S}) \times \bar{S} + P(t_a > w \times \bar{S}) \times \bar{t}_a \right\} = V_n \times \left\{ (\bar{t}_a - \bar{S}) \times e^{-\frac{S}{\bar{t}_a(\bar{t}_a - \bar{S})}} + \bar{S} \right\} \quad \dots\dots\dots (7)$$

ここで、 $V_n$  は表示すべき全直線の数  
 $P(t_a \leq w \times \bar{S})$  は  $t_a$  が  $w \times \bar{S}$  以下の確率  
 $P(t_a > w \times \bar{S})$  は  $t_a$  が  $w \times \bar{S}$  以上の確率

4. 評価結果と試作結果の考察

MP および GP の各処理ステップのクロック・サイクル時間の比を  $\alpha$  とすると式(5)~(7)は下記のようになる。

$$\bar{w} = \frac{3.52}{\alpha(\alpha - 1.875)} \quad \dots\dots\dots (8)$$

$$P(q > N) = \left(\frac{1.875}{\alpha}\right)^{N+1} \quad \dots\dots\dots (9)$$

$$T_{DISP} = 240 \times V_n \times T_{GPC} \times \left( \alpha \times e^{-\frac{6.59}{\alpha^2(\alpha - 1.875)}} + 1.875 \right) \quad \dots\dots\dots (10)$$

ただし、 $\alpha = T_{MPC}/T_{GPC}$   
 式(8)および式(10)を図示すると 図 7 のようになる。

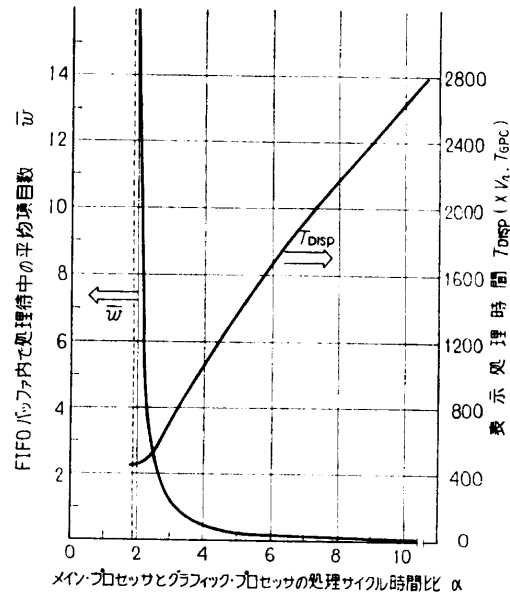


図 7 処理待中の数と表示処理時間

Fig. 7 Average number of queue and display command list processing time.

すなわち、FIFO バッファ内の平均的な待行列の数  $\bar{w}$  は  $\alpha$  が 3 以下になると急増する。一方、全コマンド・リストをスキャンし完全な表示を得るまでの表示時間  $T_{DISP}$  はすべての処理項目が FIFO バッファ内にある (FIFO バッファ容量が無制限) とし最小になり、 $\alpha$  が大きくなるにしたがって長くなる。したがって表示処理時間を短縮するためには、バッファ容量の許す範囲で  $\alpha$  を小さくすることが必要である。

必要なバッファ容量については式(9)を図示した 図 8 を参照することによって求めることができる。

すなわち、バッファがオーバーフローする確率を 1% 以下にするためには  $N=4$  であれば  $\alpha > 4.7$ 、 $N=8$  であれば  $\alpha > 3.1$  に設定する必要がある。図 1 のシステムでは 1 項目に対し平均 3 ワード (6 バイト) のデータを必要とするのでバッファ容量としては、前記項目数  $N \times 6$  バイトを準備しなければならない。今回の試作システムでは、MP として汎用の MOS 8 ビットマイコンを使用し、GP としてバイポーラのビットスライス LSI を用いた 16 ビット専用プロセッサを開発した。このため MP および GP の処理サイクル時間はそれぞれ  $T_{MP} = 1.26 \mu s$ 、 $T_{GP} = 0.20 \mu s$  に設定した。したがって  $\alpha = 6.3$  となり、FIFO バッファの容量としては  $N=4$  (すなわち 24 バイト) 以上を用意することにし、ハードウェア量とのかねあいで 32 バイトにした。この値から逆算すると、 $N=5$  となりバ

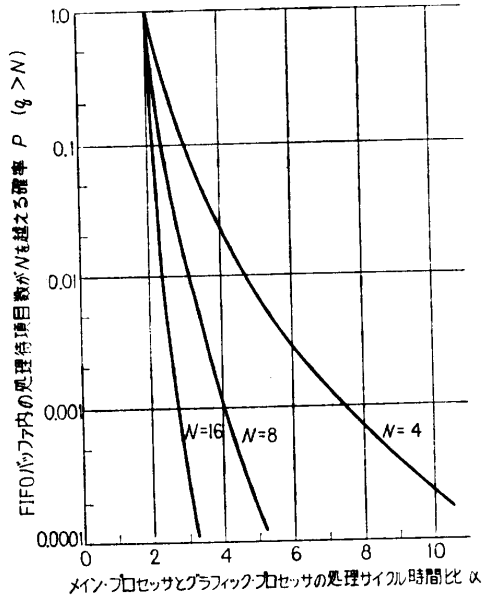


図 8 FIFO バッファがオーバーフローする確率  
Fig. 8 Probability of FIFO buffer overflow.

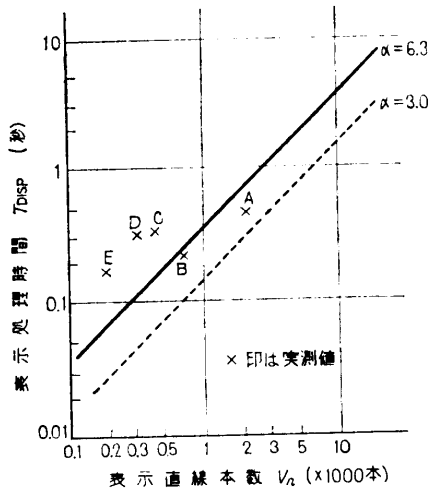


図 9 表示処理時間の計算値と実測値  
Fig. 9 Calculated value and measured value of display processing time.

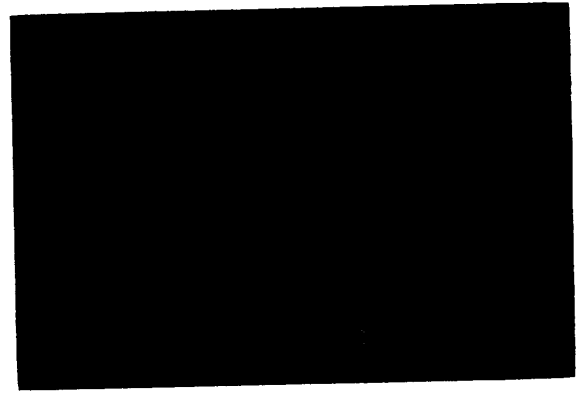


図 10 評価用画面の表示例 (画面 No. A)  
Fig. 10 Example of displayed data for evaluation.

バッファがオーバーフローする確率は 0.1% 以下となる。  
上記のような試作システムにおいて、実際にグラフィック表示をさせた場合、その画面の表示時間がどのようになるかを式(10)によって求めた結果と、いくつかの画面の実測値とを 図 9 に示す。また評価に用いた画面の概要と測定結果を 表 1 に、表示画面の例を 図 10 に示す。

表 1 より明らかなように、画面 C~E は矩形や円のぬりつぶし (面図形) が多いため、直線本数  $V_n$  で評価した場合より表示時間が長くなっている。本システムは  $\alpha=6.3$  であるため MP がフルに動作し、全体としての表示処理性能を制限している。このことは、表 1 より GP に対する処理要求項目の平均到着間隔  $\bar{t}_a$  を下記の式(11)により求めた値の平均値  $310 \mu s$  と、式(2)による  $\bar{t}_a \approx 240 \times 1.26 \mu s = 302 \mu s$  とがよく一致することからも裏付けられる。

$$t_a = \frac{T_{DISP}}{C_n + V_n} \dots\dots\dots(11)$$

したがって、システム全体の表示処理能力を向上させるには、MP と GP とのバランスをとるため MP を高速化 ( $\alpha$  を小さく) すればよく、最近の高速バージョンのマイクロプロセッサを用いれば  $\alpha=3.0$  ( $T_{MPC}$

表 1 評価用画面の測定結果

Table 1 Experimental results of evaluated display data.

画面 No.	画面名称	コマンド数 $C_n$	直線本数 $V_n$	表示時間 $T_{DISP}$	備考
A	世界地図	58 個	2,224 本	431 ms	
B	LSI セル	168	748	216	
C	立体棒グラフ	449	487	344	ぬりつぶし矩形多い
D	日本の電力量	496	363	326	円グラフ多い
E	システム構成図	320	221	194	ぬりつぶし矩形多い

$=0.6\mu\text{s}$ ) を容易に実現できる。このときの表示速度は図9の点線のようになり7,000本/秒という高速表示が達成される。ただし、これを実現するためには、バッファ容量としては、オーバフローの確率を1%としても  $N \geq 8$  すなわち48バイト以上が必要になる。

## 5. 結 言

機能分散型のデュアルプロセッサシステムをグラフィック・ディスプレイに採用する場合の問題点のひとつである両プロセッサの処理性能比が表示速度やバッファ容量に及ぼす影響を定量的に把握するため、

1) 単一サービス窓口の簡単な待行列モデルを応用し得ることを明らかにし、

2) その結果を利用して、表示速度やバッファ容量を推定する場合の設計チャートを作成し、

3) チャートによる評価データが実測値とよい対応を示すことを確認するとともに、試作システムの性能向上のための具体策を明確にした。

その結果、汎用のMOS 8ビット・マイクロコンピュータと図形発生専用の高速プロセッサという比較的単純な構成でも7,000本/秒程度の高速図形表示が可能との見通しが得られた。

しかし、本報で述べた表示性能評価手法は、表示画面内の直線本数を代表パラメータとして抽出し、かなり強引に推定している。グラフィックの応用分野の拡大ともなって面図形(いわゆるぬりつぶし図形)が多用される傾向にあり、これによる誤差が拡大するものと思われる。今後は、このような面図形処理や中間調の画像処理を含んだ、より高度な表示処理性能の評価が必要になる。

終わりに、本研究に関して指導いただいた日立製作所日立研究所川本幸雄、平沢宏太郎の両博士および本研究の製品化にご尽力いただいた日立製作所大みか工場桑原洋、林幸登、小中清司の各氏ほか多数の関係者の方々に深く感謝いたします。

## 参 考 文 献

- 1) Myers, W.: Interactive Computer Graphics-Poised for take off?, IEEE Computer, Vol. 11, No. 11, pp. 60-74 (1978).
- 2) Ware, M.: Interactive Computer Graphics-Flying High, IEEE Computer, Vol. 12, No. 7, pp. 8-17 and Vol. 12, No. 8, pp. 52-67 (1979).
- 3) Raymond, J. and Banerji, D.K.: Using a Microprocessor in an Intelligent Graphics Terminal, IEEE Computer, Vol. 9 No. 4, pp. 18-25 (1976).
- 4) 川本幸雄, 浜田長晴: ディスプレイ用 LSI, 信学誌, Vol. 61, No. 11, pp. 1239-1243 (1978).
- 5) Hoeksma, J.W.: High-resolution Interactive Colour Display, Computer-Aided Design, Vol. 9, No. 3, pp. 173-177 (1977).
- 6) Paisner, W.S.: The evolution and architecture of a high-speed workstation for interactive graphics, NCC, Vol. 48, pp. 165-173 (1979).
- 7) Weste, N. and Ackland, B.: GUMBI-A Graphic User Microprogramable Bit-slice Interpreter, COMPCON '79 Vol. CH 1465-4/79, pp. 232-237 (1979).
- 8) Shuto, T. et al.: Color Graphic Design System for IC Mask, COMPCON '79 Vol. CH 1465-4/79 pp. 280-286 (1979).
- 9) Takatoo, et al.: Software Architecture of a Graphic Display Terminal with Interactive Picture Generation Capability, 6th IFAC/IFIP Int. Conf. on DCAPC pp. 487-494 (1980).
- 10) 小林芳樹他: 高機能ディスプレイ端末のためのグラフィック・プロセッサの開発およびその評価, 情報処理学会論文誌, Vol. 22, No. 2, pp. 129-138 (1981).
- 11) Martin, J.: Design of Real-Time Computer Systems, pp. 504-505, Prentice Hall, New Jersey (1967).

(昭和55年6月20日受付)

(昭和55年12月18日採録)