

情報処理教育のための図形表示機能の実現

岡田 稔 熊谷 毅 三輪 和久 櫻井 桂一
名古屋大学情報処理教育センター

あらまし

本稿では、情報処理教育システムを指向したベクトル図形表示インターフェースの実現について述べる。著者らの所属する機関では、情報処理教育のための計算機システムとして汎用大型計算機、ワークステーション、そしてパーソナルコンピュータから構成される分散処理システムを使用している。ここで述べるインターフェースはパーソナルコンピュータ上に割り込み型の常駐プログラムとして動作するソフトウェアであるため、パーソナルコンピュータをUNIXメインフレームやワークステーションの端末エミュレータとして動作させる場合と、スタンドアロンで使用する場合の両方で動作するという特長を持つ。また、最適化されたDDA直線発生、破線発生などにより高速表示が可能である。本システムを利用することにより、専用のグラフィック・ディスプレイを必要とせず、低コストな図形表示インターフェースが実現できた。

IMPLEMENTATION OF PERSONAL GRAPHICS INTERFACE FOR COMPUTATIONAL LECTURES

Minoru OKADA, Takeshi KUMAGAI, Kazuhisa MIWA and Keiichi SAKURAI
Education Center for Information Processing, Nagoya university.
Furo-cho, Chikusa-ku, Nagoya 464-01, Japan

In this paper, we describe an implementation of a vector graphics display interface suitable for computational lectures. In our center, a distributed processing system which consists of a main-frame computer, workstations and personal computers is used for computational lectures. The interface behaves as a stay resident software driven by interruption on the personal computer, therefore it works well in either cases that the personal computer is used as a terminal or used in standalone. It is possible to draw lines fast because solid and broken lines are generated by optimized DDA. We can realize a vector graphics system at a low cost by using this interface.

1.はじめに

情報処理教育のための計算機環境では、計算機とユーザの会話環境の整備が重要である。特に計算機言語の教育を行なう場合などでは、グラフィックス機能は計算結果の可視化はもちろんのこと、受講者の興味をそそり、意欲の向上のために欠かすことのできない機能の一つであろう。

情報処理教育システムとして最近盛んに使用されるようになったUNIX*オペレーティングシステムでは、GKS,plotライブラリに代表される汎用図形作画インターフェースが提供されている。これらのインターフェースは機種に依存しない図形命令を出力すると共に、既存の汎用グラフィックターミナルをオンライン駆動する図形命令も出力することが可能である。また、GKSはUNIXのみならず多くの計算機システムに装備されている。このようにホスト計算機側の図形作画機能が充実しているにもかかわらず、他方では、それらの計算機システムを使用するための端末は、コスト削減の点から非図形端末を使用しているのが実状である。

本稿では、情報処理教育システムを指向したベクトルグラフィックス図形表示インターフェースの概要とその実装について述べる。本インターフェースは4014-1型グラフィックターミナル⁽¹⁾(以下、4014-1)の図形命令をターゲットとして、ソフトウェアエミュレーションの形でパーソナルコンピュータ上に実装される。

著者らの所属する機関では、情報処理教育のための計算機システムとして汎用大型計算機、ワークステーションおよびパーソナルコンピュータ端末を用いた分散処理システムを使用している。ここで述べるインターフェースはパーソナルコンピュータ上でソフトウェア割り込みによる常駐型プログラムとして動作するソフトウェアであるため、不特定の端末エミュレーションソフトウェアの下で動作させることが可能である。また、本インターフェースを実装するパーソナルコンピュータを、UNIXメインフレームやワークステーションの端末として動作させる場合と、スタンドアロンで使用する場合のいずれにおいても動作するという特徴がある。さらに、パーソナルコンピュータが本来持っているラスタグラフィックス表示機能を利用するため、専用のグラフ

ィック・ディスプレイやグラフィックターミナルを必要とせず、低コストなパーソナル・グラフィックス環境の実現が可能である。

2.情報処理教育における図形表示機能の問題点

情報処理教育では、実社会の計算機利用の現場における多種多様な計算機構成、ソフトウェア構成に対応できる環境を提供する必要がある。多くの企業、研究機関等では汎用大型計算機を始めとして、UNIX環境のワークステーション、MS-DOS**等を用いた汎用OS環境のパーソナルコンピュータが有機的に結合され、O.A.F.A.研究開発の環境が整備されている。大学等の情報処理教育センターにおいては、このような多様な実社会の現状に対応した情報処理教育環境を整備することが望まれる⁽²⁾。

パーソナルコンピュータを端末として通信回線によってセンターホスト計算機を利用する場合には、ホスト計算機が直接、端末のグラフィックス画面を制御することはできない。通常、パーソナルグラフィックス環境を実現するためには高価な専用のグラフィックターミナルを使用するか、図形表示機能を併せ持った端末エミュレータのもとでパーソナルコンピュータを使用する必要がある。

しかし、Ethernetなどのネットワーク環境や、複数のオペレーティングシステムに対応する必要がある教育用の計算機システムでは、特殊化された端末エミュレータの使用を強いられており、その端末エミュレータが図形表示機能を併せ持っているという保証は全く無い。従って、何らかの形で文字表示のための端末エミュレータと、図形表示のためのソフトウェアを分離できると都合がよい。

このようなことから、情報処理教育を指向した図形表示機能に対する条件として、次のような項目が挙げられる。

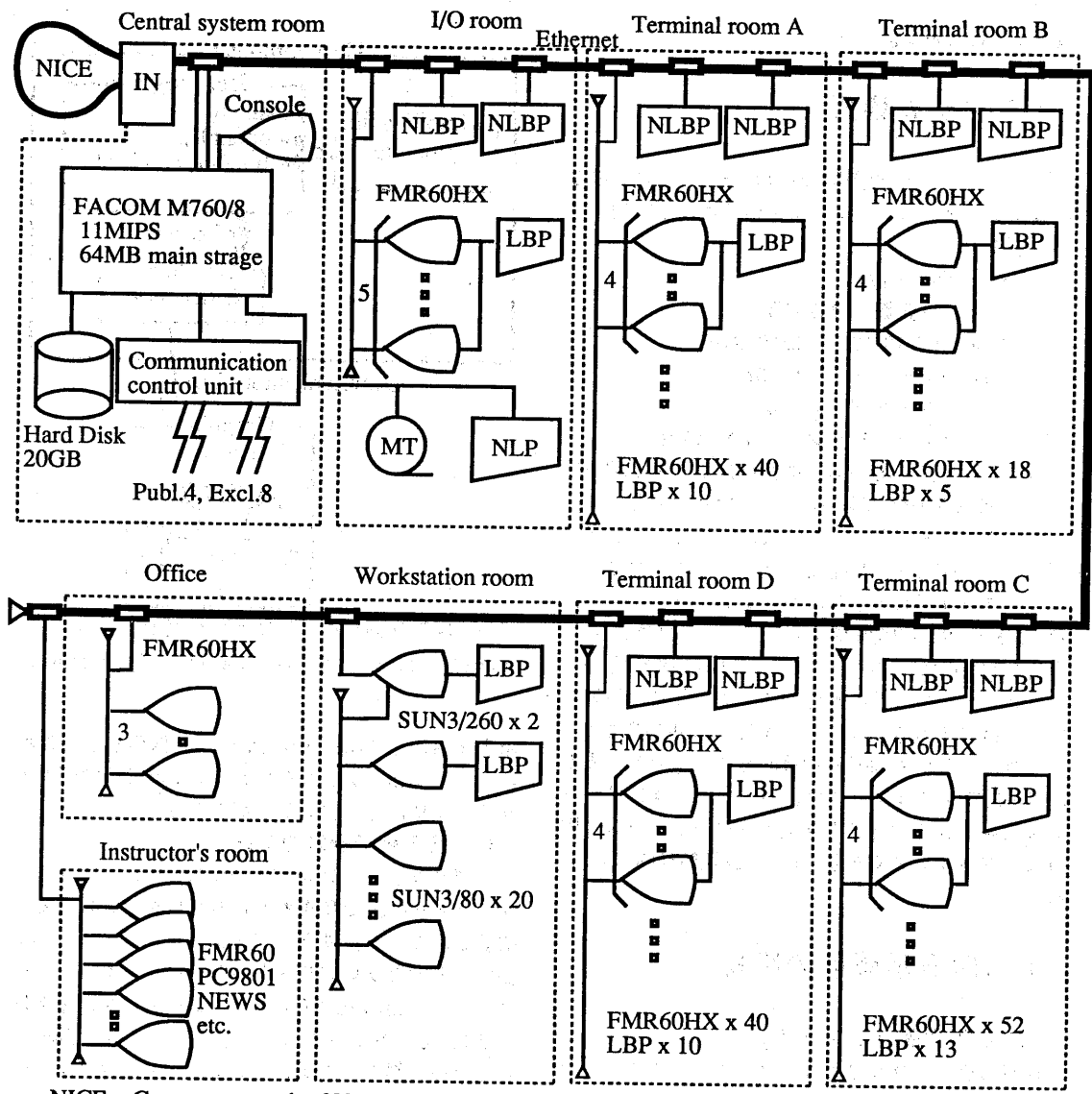
1.汎用の図形作画ライブラリパッケージの出力に対応する。

2.コスト削減のためには、文字端末とグラフィックス専用端末を別個に用意するのではなく、単一端末で相方の機能を実現するのが望ましい。

3.既存の端末エミュレータプログラムから独立した形態が望ましい。

本文で述べる図形表示システムはこれらの要求を

*UNIXは米国ベル研究所の登録商標である **MS-DOSは米国マイクロソフト社の登録商標である



NICE : Campus network of Nagoya university.
 NODE : Optical network node.
 NLBP : Network laser beam printer.
 NLP : Japanese-printable line printer.
 LBP : Laser beam printer.

図1 名古屋大学情報処理教育センター計算機システム構成図
 Fig.1 A computer system at Education Center for Information Processing, Nagoya university.

満たすものである。

3. システム構成

現在の名古屋大学情報処理教育センターにおける計算機システムのハードウェア構成は以下の通りである[図1]。

[ハードウェア構成]

ホスト計算機 : M760/8

端末 : FMR-60HX 20MB 160台

端末用レーザプリンタ 40台

ネットワークプリンタ 4台

ワークステーション : SUN3 22台

ホスト計算機M760/8は、専用OS(MSP)と汎用OS(UTS/UNIX System V系)のデュアルOSのもとで動作している。ユーザが使用する端末は専用端末とはせず、EthernetによりLAN接続されたパーソナルコンピュータを使用している。パーソナルコンピュータは、MS-DOS、UTS用端末エミュレータプログラムを含むTCP/IPソフトウェア群、およびMSP用端末エミュレータのもとで動作している。ここで使用しているMSP用端末エミュレータは、MSP独自のプロトコルに基づく図形表示機能を有している。また、UTS用端末エミュレータは、パーソナルコンピュータに内蔵されるLANアダプタインターフェースに固有なソフトウェアであり、図形表示機能は有していない。

4. 図形表示機能の実現

4.1. 図形作画ライブラリ

図形作画ライブラリとは、グラフィックターミナルを駆動する図形命令を出力するためのソフトウェア群である。ユーザは文字列として構成されている図形命令をグラフィックターミナルに送ることにより、CRT上に図形を表示することができる[図2]。

UNIXオペレーティングシステムを始めとして、GKS系の汎用図形作画ライブラリが充実して来ており、また、GKS系の図形命令セットを既存のグラフィックターミナル用図形命令セットに変換するフィルタ群も充実している。既存のグラフィックターミナルの中でも、4014-1は古典的でありながら、事実上現在の図形作画ライブラリのための標準端末とされている。そこで今回作成する図形表示システムの

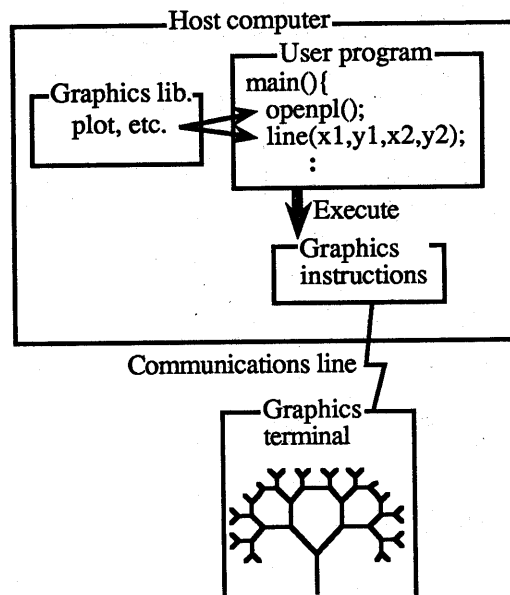


図2 図形作画ライブラリと図形表示

Fig.2 Graphics libraries and a method of figure drawing.

命令体系として4014-1の命令体系を採用した。これにより、GKS,plotを始めとする多くの汎用図形作画ライブラリに対応することができる。

4.2. 図形表示機能の実装

本図形表示システムはMS-DOSパーソナルコンピュータとその上で動作する制御プログラムによって構成されている。以下、パーソナルコンピュータで動作する制御プログラムを図形表示インターフェースという。図形表示インターフェースは主記憶空間に常駐⁽⁹⁾してソフトウェア割り込みによって駆動されるため、端末エミュレータなどの主アプリケーションソフトウェアの動作にはほとんど影響を与えず、共存が可能である。

4.3. MS-DOSとのインターフェース

MS-DOSでは、アプリケーションソフトウェアとOSのインターフェースとしてシステムコール⁽⁹⁾と呼ばれるソフトウェア割り込みを使用している。画面に対するテキスト文字出力においても標準化のためにシステムコールを利用している端末エミュレータが多い。この文字出力のためのシステムコールは複数存在するが、それらはすべてINT29と呼ばれる非公開な内部的ソフトウェア割り込みを経由している。

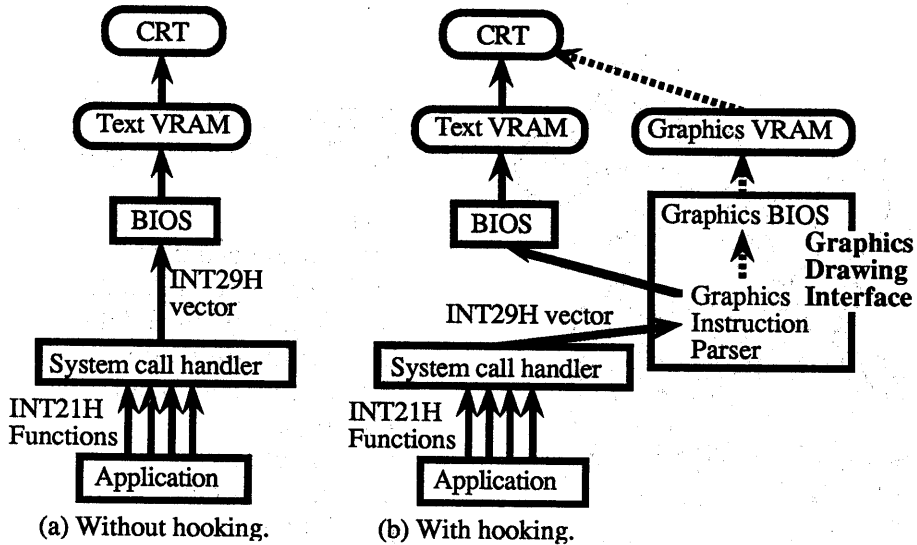


図3 割り込みのフッキングによるインターフェースの実現
Fig.3 Implementation of graphics interface by hooking of interrupt.

これは筆者が調査した限り、すべてのMS-DOSパーソナルコンピュータにおいて同様であった。INT29割り込みが生じると、通常(図形表示インターフェースが常駐していない状態)は文字描画のためのBIOSに制御が移る。この動作は割り込みベクトルがBIOSに向いているといわれる。

一方、本インターフェースを常駐させると、先のINT29割り込みベクトルは書き換えられ、インターフェース内の処理部に向けられる。処理部の後部には元のBIOSに制御を移すための命令が置かれている。常駐後、INT29割り込みが生じるとインターフェース内部の処理部を経由してBIOSが動作することになる。この処理部では後に述べる図形命令の字句解析、構文解析、図形命令の実行が行なわれ、図形命令でない文字はBIOSに送られる。

これらの処理過程の様子を図3に示す。なお、常駐時に元のINT29割り込みベクトルを退避しておき、開放時に復元させれば図形表示インターフェースの影響を残すことなく終了させることができる。

4.4. 図形命令の解析

図形命令は、1文字または複数文字から構成される文字列であり、通常のテキスト表示文字に混入されてホスト計算機から送出される。このため、端末側ではこれらの図形命令文字と非図形命令文字の分離を行ないながら、図形命令の解析および実行を行

なう必要がある。4014-1図形命令の仕様に基づいて作成した状態遷移図を図4に示す。本図形表示インターフェースでは、この状態遷移に基づいて図形命令の構文解析、字句解析を行なっている。

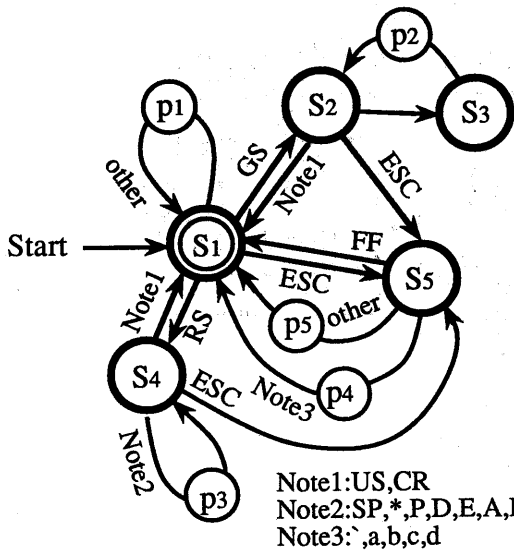
なお、本図形表示インターフェースでは4014-1図形命令を使用したことによって、一部の文字、文字列は図形命令として優先的に解釈され、テキスト画面に対して反映させることはできない。これらは状態遷移図中の S_1 から出るGS*,RS*およびESC*で始まる文字列である。但し、ESCで始まる文字列のうち、 S_1 に終端されない文字列は図形命令ではないと判断され、テキスト画面に反映される。GS,RSは多くの端末制御シーケンスでカーソル移動命令とされており、UNIX等の端末として使用する場合に不都合が起きるが、GS,RSを使用せずともカーソル移動が実現できるため、この不都合は容易に回避可能である。

4.5. DDAによる線分発生と破線発生

ラスタ型ディスプレイを用いて線図形を表示させるときに基本となる処理は、1画素の描画とそれを用いたデジタル線分発生である。ここではデジタル微分解析(Digital Differential Analyser)に基づくBresenhamの方法⁶⁾を用いて線分を表示している。

ところで、単純な直線分や鎖線を始めとして多くの種類の破線を選択的に発生させるためには、複雑な処理を必要とするためにコストが大きくなりがち

*GS,RS,ESCなどはASCIIコード制御文字の名称である



$M=(Q, \Sigma, \delta, q_0, F)$
 $Q=\{S_i\}, \Sigma=\{\text{ASCII codes}\}, q_0=S_1, F=\{S_1\}$

States

- S1: Waiting characters
- S2: Waiting point A data
- S3: Waiting point B data
- S4: Waiting direction
- S5: Waiting command

Functions

- p1: Text character
- p2: Draw a vector
- p3: Plot a point
- p4: Change line style
- p5: output ESC+character

Note1: US, CR
 Note2: SP, *, P, D, E, A, I, H, J, B, P
 Note3: ` , a, b, c, d

図4 図形命令パーザの状態遷移図

Fig.4 State transition diagram in the graphics instruction parser.

Hex	B3B2B1B0	B1	B2+B1	B3+B1	B3+B2+B1
0	0 0 0 0	0	0	0	0
1	0 0 0 1	0	0	0	0
2	0 0 1 0	1	1	1	1
3	0 0 1 1	1	1	1	1
4	0 1 0 0	0	1	0	1
5	0 1 0 1	0	1	0	1
6	0 1 1 0	1	1	1	1
7	0 1 1 1	1	1	1	1
8	1 0 0 0	0	0	1	1
9	1 0 0 1	0	0	1	1
A	1 0 1 0	1	1	1	1
B	1 0 1 1	1	1	1	1
C	1 1 0 0	0	1	1	1
D	1 1 0 1	0	1	1	1
E	1 1 1 0	1	1	1	1
F	1 1 1 1	1	1	1	1

Hex-mask	pattern	
no-care	Solid	
0002H	Dotted	
0006H	Short dash	
000AH	Dot-dash	
000EH	Long dash	

(a) Truth table of Boolean operations for 4-LSBs of DDA counter.

(b) The masks for DDA-counter and produced line patterns.

図5 ビット演算による高速破線発生

Fig.5 Fast broken line generation by bit operations.

である。本図形表示インターフェースでは以下に述べるビットパターン法によって5種類の破線を発生させている。

まず、描画位置に基づき整数値を決定する。実際にはこの値としてDDA内で使用されるカウンタの下位4ビットを用いている。2進数表現された整数値のビットパターンの複数の特定ビットの論理和が真のときに当該画素を表示し、偽のときは非表示とする。論理和の対象となるビットを変更すると、それに応じて破線パターンが変化する。この処理をDDAでの逐次的な画素描画と併用することによってさまざまな線種の破線が得られる[図5]。

この破線発生方法は先に述べたDDAによる線分発生方法と親和性が良く、DDAアルゴリズム中に容易に挿入可能である。また、実際の処理プログラムにおける破線判定では、ワード単位の並列論理演算を用いて行なわれるために低コストとなっている。なお、以上のDDA直線発生、破線発生およびウィンドウクリッピング⁷⁾は全て整数演算に基づくものである。これにより高級言語で実装したにもかかわらず、高速表示および、浮動小数点演算ルーチンの排除によるオブジェクトの小型化が可能となっている。

4.6. 図形表示と実行時間

本図形表示インターフェースを用いて描画した図形を図6~8に示す。図6,7はよく知られている再帰曲線である。図8は本図形表示インターフェースで生成可能な破線の線種を全て含む図形である。表1にはそれぞれの図形における表示経過時間を示す。表中、左欄はEthernet(TCP/IP,10Mbps)で接続されたホスト計算機M780/8上で図形命令を出力するプログラムによりオンラインでリモート表示したときの経過時間である。右欄は同プログラムにより出力された図形命令セットのファイルをパーソナルコンピュータのハードディスク上に置き、MS-DOSのtypeコマンドにて最も理想的な状態でローカル表示したときの経過時間である。

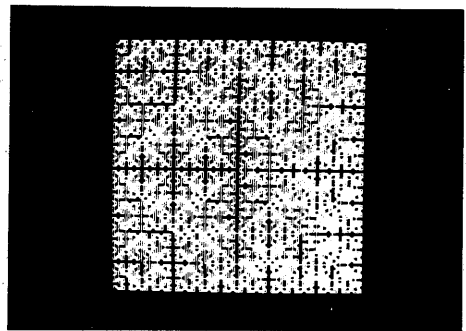
5. まとめ

本稿では、情報処理教育システムを指向したベクトル図形表示インターフェースの概要と、そのパーソナルコンピュータ上への実装について述べた。著者らの所属する機関では、情報処理教育のための計算機システムとして汎用大型計算機,ワークステー

表1 表示所要時間

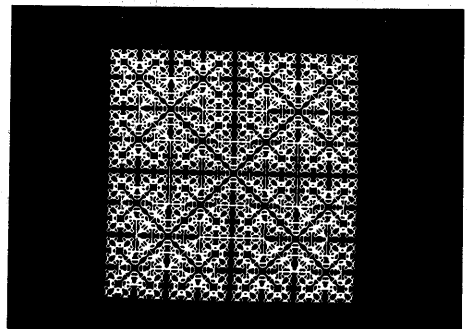
Table 1. Drawing time.

TIME [sec]	Hilbert (Fig.5)	Sierpinski (Fig.6)	Broken-lines (Fig.7)
PC9801VX2	23 11	25 11	3.5 2
FMR60HX	11 5	11 5	2 1



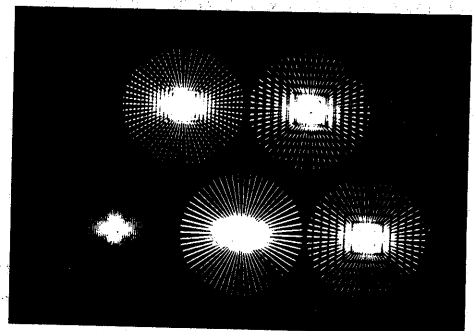
[図6] 5レベルのヒルベルト曲線群

Fig.6 Hilbert curves consist of 5 recursive levels.



[図7] 4レベルのシェルピンスキー曲線群

Fig.7 Sierpinski curves consist of 4 recursive levels.



[図8] 破線を含む直線群

Fig.8 Broken lines.

ション,およびパーソナルコンピュータから構成される分散処理システムを使用している。このような分散処理環境においては、分散処理を指向したグラフィックス環境が望まれる。パーソナルコンピュータ上に実装された本図形表示インターフェースは次のような特徴を持っている。

(1)専用のグラフィック・ディスプレイやグラフィックターミナルを必要とせず、低コストなベクトルグラフィックス環境を実現できる。

(2)割り込み型の常駐プログラムであるため、端末エミュレータなど、不特定の主アプリケーションの起動中でも動作する。

(3)4014-1型グラフィックスターミナルの作画命令をターゲットとしたため、多くの図形作画ライブラリに対して汎用性がある。

(4)整数演算に基づくDDAによる直線発生、最適化された破線発生、ウィンドウクリッピングにより、高速な図形表示が可能である。

現在、図形表示インターフェースとしての基礎的機能が完成している。カラー図形の表示およびシステムコールを使用していない端末エミュレータへの対応が課題となっている。

謝辞

本研究に当たり、ご指導頂いた名古屋大学情報処理教育センター長鬼頭幸生教授に深謝する。なお、本研究の一部は文部省科研費（一般研究(B)No.01450270）による。

参考文献

- (1)"4014-1型グラフィック・コンピュータ・ターミナル インストラクション・マニュアル",ソニー・テクトロニクス
- (2)情報処理教育に関する研究会各論文, 国立大学情報処理教育センター協議会,(1990)
- (3)星野操:"MS-DOSレジデントプログラム入門", 技術評論社,(1989)
- (4)MS-DOS3.1プログラマーズリファレンスマニュアル,Microsoft&NEC,(1985)
- (5)"富士通FMRシリーズ徹底解析マニュアル",インタープログ編,ビー・エヌ・エヌ,(1987)
- (6)Bresenham,J.E., "Algorithm for Computer Control of Digital Plotter", IBM Syst.J.,4(1),pp.25-30,(1965).
- (7)Sproull,R.F. and Sutherland,I.E., "A Clipping

Divider", pp.765-775, FJCC, Thompson books, Washington, D.C., (1968).

付 録

[図形表示インターフェースの仕様]

コマンド名

tekplot.exe

動作機種

FMRシリーズ (1120 x 750 pixels)

PC9801シリーズ (640 x 400 pixels)

動作OS

MS-DOS2.1以上

起動の方法

tekplot [-r][-c][[-s<figure>]][<filename>] <cr>

起動オプション

-r: プログラムの解放

-c: グラフィック画面消去

-s: xy座標値の右シフトビット数

ファイルを指定するとそのファイルの内容を図形命令として図形表示する。ファイルを指定しない場合はメモリに常駐し、MS-DOSの画面出力システムコールをフックしてグラフィック端末エミュレータとして動作する。

オプション[-r]は、常駐しているプログラムを開放する。オプション[-c]は画面を消去する。オプション[-s]とそれに続く1桁の数字は座標値の右シフトビット数を指定する。これはパーソナルコンピュータの表示ドット数よりも大きな仮想座標に基づいて描かれた図形の表示時に有効である。4014-1図形端末の仮想座標系は(4096×4096)の大きさを持つ。もちろん、1ビット右シフトは仮想座標値を1/2にして表示する。

記述言語

C

処理系

Turbo C 2.0

(FMR用はクロスコンパイルによる)

オブジェクトサイズ

PC9801シリーズ用: 13,270Bytes

FMRシリーズ用: 13,384Bytes

常駐時占有メモリ

PC9801シリーズ用: 21,744Bytes

FMRシリーズ用: 21,856Bytes