

〔非公開〕

TR-C-0061

ステレオカラー動画処理装置 GRAPHICA  
ユーティリティー マニュアル

石橋 聡  
Satoshi Ishibashi

1991. 2. 8

A T R 通信システム研究所

GRAPHICA FMU-2088SP  
ユーティリティ マニュアル

石 橋 聡

1991年2月4日

ATR通信システム研究所

本書はステレオカラー動画処理装置 GRAPHICA FMU-2088SPを操作するためのユーティリティーに関する説明書である。

本書は以下により構成される。

### 本書の構成

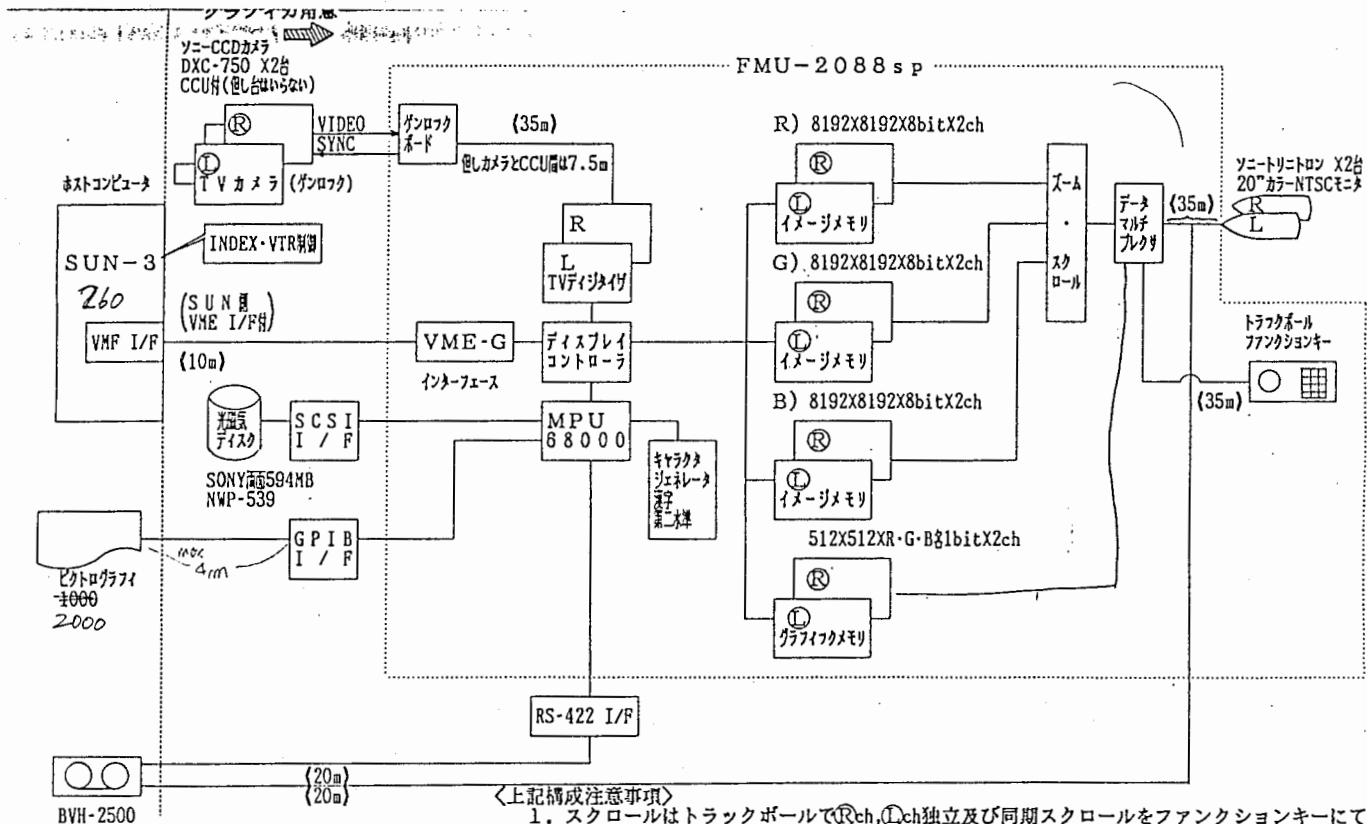
- ・ GRAPHICA FMU-2088SP概要
- ・ 対話型コマンド実行ツール (1) GRA
- ・ 対話型コマンド実行ツール (2) GIM
- ・ GRAPHICA ユーティリティー説明
- ・ 付録  
ユーティリティーソースリスト

# GRAPHICA FMU-2088SP 概要

装置構成及び機能を以下に示す。

品名	性能	機能	数量
動画像処理装置 FMU-2088SP	<ul style="list-style-type: none"> <li>・表示分解能 512×512dot</li> <li>・イメージメモリ 8192X8192dotXR・G・B各8bitX2ch</li> <li>・グラフィックメモリ 512X512XR・G・B各1bitX2ch</li> <li>・マイクロプロセッサ MC68000</li> <li>・CRT20インチカラー</li> <li>・ズーム ×2, ×4, ×8</li> <li>・スクロール Rch・Lch 同期・非同期可 ×64dot, y1dot</li> <li>・キャラクタジェネレータ 漢字第二水準</li> <li>・カラーモニタ 21インチ NTSC</li> <li>・トラックボール</li> <li>・ファンクションキー</li> </ul>	イメージメモリの 入出力制御・表示 制御を行う	1
ホストコンピュータ インターフェース IF-VME-G	SUN標準 VME規格に準ずる	制御コマンドの 転送	1
CCDカラーTVカメラ DXC-750	水平解像度 700本 有効画素数 38万画素 S/N比 60dB カメラコントロールユニット 15倍ズームレンズ	画像入力	2
光磁気ディスク NWP-539	容量 両面594MB (片側297MB) 使用メディア カートリッジ型(130mm) インターフェース SCSI	画像記録	1
ピクトグラフィ インターフェース IF-GPIB-G	フジフィルム標準 GPIBに準ずる	画像出力制御コマ ンドの転送	1
VTR インターフェース IF-RS-422-G	SONY標準 RS-422に準ずる	VTR制御コマ ンドの転送	1

ハードウェア構成を以下に示す。



＜上記構成注意事項＞

1. スクロールはトラックボールでⓇch, Ⓛch独立及び同期スクロールをファンクションキーにて切り替える(ホストからも制御可能)。
2. ピクトロハードコピーで512x512dotの画像を1枚全体に出す機能と4つを1枚に出す機能を持つ。
3. 光磁気ディスクは片ch全領域を片面で記録できる(片面297MB, 書き込み可能)。さらに片chずつ(L/R)の一部の領域を光磁気ディスクの片面に記録することもできる。

## ソフトウェアの所在

ソフトウェアは以下の場所に格納されている。

CS05: /usr/graphica

CS05: /usr/graphica/local/src

CS05: /usr/graphica/local/GIMsrc

CS05: /usr/graphica/local/bin

## プログラミング上の留意点

グラフィカのライブラリは FORTRAN で記述されている。  
そのため以下の点に留意すること。

### 1) C言語からコールする場合

サブルーチン名\_(パラメータ)



必ずこれを付ける。

### 2) リンクの方法

%cc ソース名 -O実行モジュール名 -lgisp -IF77 -II77 -IU77

## 対話型 コマンド実行ツール (1)

### GRA

gra

gra-pic

gra-mo1

gra-mo2

gra-gio



GRAは (株)グラフィカがテスト用に開発した  
コマンド対話ツールである。

- ・ gra ----- 通常の処理コマンド
- ・ gra-pic ----- ピクトログラフィー用処理コマンド
- ・ gra-mo1 ----- 光ディスク用処理コマンド
- ・ gra-mo2 ----- 光ディスク用処理コマンド
- ・ gra-gio ----- グラフィック系処理コマンド

#### 使用法:

コンソールから上記モジュール名をタイプインすると  
メニューが表示されるので、これに従い操作すればよい。

テスト用であるのでユーザインタフェースは悪いが、パラメータまで対話形式で入力できるので、入門レベルで使用するのに適している。

対話型コマンド実行ツール (2)

GIM

Graphica Command Interaction Module

GIMはATRで開発したコマンド対話ツールであり、VICOMのcim同様処理サブルーチン単位で対話形式で実行できる。

使用法:

% gim

\*>

← gimのプロンプト

\*> end

終了

%

詳細は gim のマニュアル参照

尚、ソース、実行モジュールはそれぞれ

CS05:/usr/graphica/local/GIMsrc

CS05:/usr/graphica/local/bin

に格納されている。

## G I M の 機 能 概 要

システム名称 GLSPコマンドインタプリタ

システム略称 G I M

作成者

CSK

作成日

90-04-30

改訂

本システム (G I M) はステレオ動画フレームメモリ Graphica FMU-2088SP を  
ホストコンピュータ (sun3/160) より対話形式で動作させるための実行モジュール形式の  
ソフトウェアでありシェルから起動することにより動作します。

G I M コマンドは 1 行 1 個のコマンド文字列により構成されコマンド文字列はニューモニックと  
パラメータの 2 つの部分で構成されています。

【CMD】      【PARAM】

【CMD】      : 頭文字 3 文字以上のニューモニック

【PARAM】    : , で区切ったパラメータの並び  
但し、' , ' の後にすぐ、' がきたらデフォルト値とみなす。

G I M コマンドの実行は、キーボードから直接入力する方法と、後述のプロシージャファイルから、

読み込む方法の 2 種類があります。

## G I M の 使 用 方 法

システム名称	GLSPコマンドインタプリタ					
システム略称	GIM	作成者	CSK	作成日	90-04-30	改訂

まずはじめに、グラフィックディスプレイ装置、光ディスク装置等の付属ハードウェアの電源を入れておく等、

あらかじめ準備をしておきます。

次に、シェルのコマンド入力待て以下のようにタイプし、リターンキーを押します。

```
g i m
```

すると以下のようなプロンプトが表示され、GIMコマンドのキー入力待ちになります。

```
* >
```

注： \*の個数は現在の階層を表し、callコマンド（プロシージャファイルの実行）が実行されるたびに1個ずつ増加します。（例： \*\*> ）  
\*が2個以上のときキーボードからリターンキーのみを押すと\*が1個減ります。

GIMパラメータには、文字列、数値（10進数）、HEX（16進数）の3種類があり、省略可能なパラメータ（操作説明書参照）を省略するとデフォルト値が使用されます。下表にパラメータごとの使用例を示します。

パラメータ種類	通常例	省略例
文字列	MOINI 'MediaTitle' ' ' (シングルクォーテーション)で文字列を囲む	文字列 は す べ て 省 略 不 可
HEX	KANJI 0, 3021, 3022 '亜' '唾'	KANJI , 3021, 3022 (注: 0を省略した)
数 値	TVOUT 0 NTFMP 1, 1, 1, 1	TVOUT (注: 0を省略した) NTFMP , , 1, 1 (注: 1を2個省略した)

## プロシージャファイル

システム名称	GLSPコマンドインタプリタ					
システム略称	GIM	作成者	CSK	作成日	90-04-30	改訂

プロシージャファイルとは、GIMコマンドの集まりをファイルにしたもので、".gc" を拡張子としてもつ

ファイル名で識別されます。プロシージャファイルの1行は1つのGIMコマンドになっており、

call 'プロシージャファイル名' で実行できます。(操作説明書の説明参照)

プロシージャファイルの便利な点は必ず行わなければならない同じ流れの処理をあらかじめプロシージャファイル

の中に記述しておけばそれ以後callコマンドを用いることにより簡単に何回でも実行できることです。

プロシージャファイルでのみ使用できるコマンドとして下表のものがあり、これらのコマンドは

GIMコマンドの実行の流れを制御する命令です。

なお、プロシージャファイルの実行はEOF(ファイルの終端)の検出または、エラー発生によって終了します。

コマンド名	機 能
REPEAT [繰り返し数]	REPEAT END までの コマンド行を指定の繰り返し数だけ繰り返す。
PAUSE	キーボードからGIMコマンドの入力実行を行う。
END	プロシージャファイルの実行を終了する。

プロシージャファイルの例

ファイル名: 'sample.gc'

```

VCOLR 150, 70, 70
REPEAT 3
    TVOUT 0
    DCIRC 100, 100, 50
    PAUSE
    TVOUT 1
    PAUSE
END REPEAT
END
```

コマンド	機 能 【 使 用 例 】	パ ラ メ ー タ			
		No	デ-ータ型	省略値	内 容
OPNGD	デバイスのオープン 【OPNGD】				パラメータなし
CLSGD	デバイスのクローズ 【CLSGD】				パラメータなし
MEMSL	書き込みメモリ選択 【MEMSL 7】	1	整数	7	メモリ選択番号(0~7)
GERS	MEMORY消去 【GERS】				パラメータなし
PAGSL	ページ選択 【PAGSL 0】	1	整数	0	ページ番号(シングル:0-7、ステレオ:0-3)
NTFMP	NTSC画像入力 パラメータセット 【NTFMP 1,1,1,1】	1 2 3 4	整数 整数 整数 整数	1 1 1 1	画像サイズ番号 撮り込みフレーム間隔 撮り込み開始フレーム番号 撮り込みフレーム数
CMFMG	カメラ画像入力起動 【CMFMG】				パラメータなし
VTFMG	VTR画像入力起動 【VTFMG 0,0,0,0】	1 2 3 4	整数 整数 整数 整数	0 0 0 0	時間 (0~23) 分 (0~59) 秒 (0~59) フレーム (0~29)
VTRPC	カメラ画像録画 パラメータセット 【VTRPC 0,0,0,0, 0,0,0,0,】	1 2 3 4 5 6 7 8	整数 整数 整数 整数 整数 整数 整数 整数	0 0 0 0 0 0 0 0	録画開始時間 録画開始分 録画開始秒 録画開始フレーム 録画終了時間 録画終了分 録画終了秒 録画終了フレーム
VTRGC	カメラ画像録画起動 【VTRGC】				パラメータなし
VTRCMD	VTR基本動作制御 【VTRCMD】	1	整数	5	動作番号 1:REC,2:PLAY,3:FF, 4:REW,5:STOP
CINEI	フレーム画像連続表示・ 録画パラメータセット 【CINEI 1,1,1,1,1】	1 2 3 4 5	整数 整数 整数 整数 整数	1 1 1 1 1	画像サイズ番号 表示スピード(1~999) 開始フレーム番号 表示フレーム数 繰り返し回数(1~255)

コマンド	機 能 【 使 用 例 】	パ ラ メ ー タ			
		No	データ型	省略値	内 容
CINEG	シネループ起動 【CINEG】				パラメータなし
VTRGF	フレーム画像録画起動 【VTRGF 0,0,0,0】	1 2 3 4	整数 整数 整数 整数	0 0 0 0	時間 (0~23) 分 (0~59) 秒 (0~59) フレーム (0~29)
STCINE	シネループ・キャンセル 【STCINE】				パラメータなし
CMRSEL	入力カメラ選択 【CMRSEL 0】	1	整数	0	入力カメラ (0:Rカメラ,1:Lカメラ)
SYNCM	シングル・ステレオモード設定 【SYNCM 0】	1	整数	0	モード (0:シングル,1:ステレオ) <del>コンパニオン</del> <span style="border: 1px solid black; border-radius: 50%; padding: 2px;">CMRSEL</span> を実行
VRECSL	VTR録画メモリー選択 【VRECSL 0】	1	整数	0	選択フレームメモリ (0:Rチャンネル 1:Lチャンネル)
SMEMRL	メモリーチャンネル選択 【SMEMRL 0】	1	整数	0	選択フレームメモリ (0:R/Lチャンネル 1:Lチャンネル 2:Rチャンネル)
FRMTVO	TV出力(フレーム単位) 【FRMTVO 0,1】	1 2	整数 整数	0 1	表示モード(0:ストップ 1:スタート) フレーム番号(シングル1-512,ステレオ1-256)
INITL	画面の初期化 【INITL】				パラメータなし
VCOLR	ベクトル色指定 【VCOLR 255,255,255】	1 2 3	整数 整数 整数	255 255 255	赤の濃淡 (0~255) 緑の濃淡 (0~255) 青の濃淡 (0~255)
VKIND	ベクトル線種指定 【VKIND 0】	1	整数	0	線種 (0~7)
VECSL	ベクトル切り替え 【VECSL 0】	1	整数	0	0:スムーズベクトル 1:ノンスムーズベクトル
DLINA	折れ線表示・絶対座標 【DLINA 0,0,100,100...】	1 2 3 4 . . . . .	実数 実数 実数 実数 . . . . .	— — — — . . . . .	1点目のX座標 1点目のY座標 2点目のX座標 2点目のY座標 3 " X座標 3 " Y座標 . . 以下X, Yの繰り返し



コマンド	機 能 【 使 用 例 】	パ ラ メ ー タ			
		No	デ-タ型	省略値	内 容
DLINR	折れ線表示・相対座標				DLINAと同じ
ELINA	折れ線消去・絶対座標				DLINAと同じ
ELINR	折れ線消去・相対座標				DLINAと同じ
DCIRC	円表示・絶対座標 【DCIRC 10,10,5】	1 2 3	整数 整数 整数	— — —	円の中心のX座標 円の中心のY座標 円の半径
ECIRC	円消去・絶対座標				DCIRCと同じ
DARC	円弧表示・絶対座標 【DARC 10,10,50,359】	1 2 3 4 5	整数 整数 整数 整数 整数	— — — — —	円の中心のX座標 円の中心のY座標 円の半径 始点の角度(0~359) 終点の角度(0~359)
EARC	円弧消去・絶対座標				DARCと同じ
PRECT	矩形塗りつぶし表示 【PRECT 0,0,50,100】	1 2 3 4	整数 整数 整数 整数	— — — —	矩形の始点(左下角)のX座標 矩形の始点(左下角)のY座標 矩形の幅 矩形の高さ
ERECT	矩形塗りつぶし消去				PRECTと同じ
PPOLY	多角形の塗りつぶし表示 【PPOLY 0,0,100,100...】	1 2 3 4 . . . . .	整数 整数 整数 整数 . . . . .	— — — — . . . . .	1点目のX座標 1点目のY座標 2点目のX座標 2点目のY座標 3 // X座標 3 // Y座標 . . 以下X, Yの繰り返し
EPOLY	多角形の塗りつぶし消去				PPOLYと同じ
VCTAE	キャラクタ列表示 【VCTAE 'abcde'】	1	文字列	''	表示する文字列
TVOUT	TVへの出力起動/停止 【TVOUT 0,0,0】	1 2 3	整数 整数 整数	0 0 0	1: 起動 0: 停止 ディスプレイ上の領域の左下点X ディスプレイ上の領域の左下点Y

コマンド	機 能 【 使 用 例 】	パ ラ メ ー タ			
		No	データ型	省略値	内 容
KANJI	漢字表示  【 KANJI 0,2122,2123,.. .....】	1 2 3 4 . . .	整数 HEX HEX HEX . . .	0 — — — . . .	1 : 縦方向表示 0 : 横方向表示 1文字目の漢字コード (16進数) 2文字目の漢字コード (16進数) 3文字目の漢字コード (16進数) . . .
PAGIN	表示ページ番号の表示 【 PAGIN】				パラメータなし
FDLIN	折れ線表示・絶対座標  【 FDLIN 1,1,1,0,x1,y1, x2,y2,.....】  但しx1,y1,x2,y2...は 座標値	1 2 3 4 5 6 7 8 . . .	整数 整数 整数 整数 整数 整数 整数 整数 . . .	1 1 1 0 — — — — . . .	カラーモード 1 : フルカラー フレーム番号 1 : 512×512 フレーム番号 モード 0:シングル 1:ステレオ 1点目のX座標 1点目のY座標 2点目のX座標 2点目のY座標 . . . 以下X, Yの繰り返し
FELIN	折れ線消去・絶対座標				FDLINと同じ
FPPLY	多角形塗りつぶし表示				FDLINと同じ
FEPLY	多角形塗りつぶし消去				FDLINと同じ
FPRCT	矩形塗りつぶし表示  【 FPRCT 10,10,20,30, 1,1,1,0】	1 2 3 4 5 6 7 8	整数 整数 整数 整数 整数 整数 整数 整数	— — — — 1 1 1 0	矩形の始点のX座標 矩形の始点のY座標 矩形の幅 矩形の高さ カラーモード 1 : フルカラー フレーム番号 1 : 512×512 フレーム番号 モード 0:シングル 1:ステレオ
FERCT	矩形塗りつぶし消去				FPRCTと同じ
FCHAR	文字列の表示  【 FCHAR 100,100,'ABCDE', 1,1,1,0】	1 2 3 4 5 6 7	整数 整数 文字列 整数 整数 整数 整数	— — " 1 1 1 0	文字表示座標X 文字表示座標Y 表示文字列 カラーモード 1 : フルカラー フレーム番号 1 : 512×512 フレーム番号 モード 0:シングル 1:ステレオ

コマンド	機 能 【 使用例 】	パ ラ メ ー タ			
		No	データ型	省略値	内 容
OPNPCT	ビクログラフイーのオフン 【OPNPCT】				パラメータなし ビクログラフイーの場合、最初に実行すると
BRIGHT	DARK/BRIGHTの設定/表示 【BRIGHT 0,0】設定 【BRIGHT 1】設定値表示	1 2	整数 整数	1 0	0:設定 1:設定値表示 DARK/BRIGHT状態 0:NORMAL 1:DARK 2:BRIGHT
NEGPOS	ネガ/ポジの設定/表示 【NEGPOS 0,0】設定 【NEGPOS 1】設定値表示	1 2	整数 整数	1 1	0:設定 1:設定値表示 ネガポジ状態 0:ネガデータ 1:ポジデータ
DTTYP	データタイプの設定/表示 【DTTYP 0,0】設定 【DTTYP 1】設定値表示	1 2	整数 整数	1 1	0:設定 1:設定値表示 データ・タイプ 0:カラーコート 1:RGBコート
WNDVIW	ウィンドウビューポートの 設定/表示 【WNDVIW 0,0,0,1023,1279 ,0,0,1023,1279】設定 【WNDVIW 1】設定値表示	1 2 3 4 5 6 7 8 9	整数 整数 整数 整数 整数 整数 整数 整数 整数	1 0 0 1023 1279 0 0 1023 1279	0:設定 1:設定値表示 M100側イメージ・エリア始点X座標 M100側イメージ・エリア始点Y座標 M100側イメージ・エリア終点X座標 M100側イメージ・エリア終点Y座標 ビューポート・エリア始点X座標 ビューポート・エリア始点Y座標 ビューポート・エリア終点X座標 ビューポート・エリア終点Y座標
MAGSIZ	プリントサイズの設定/表示 【MAGSIZ 0,1,1】設定 【MAGSIZ 1】設定値表示	1 2 3	整数 整数 整数	1 1 1	0:設定 1:設定値表示 拡大・縮小要因その1 拡大・縮小要因その2
PRINT	プリントの開始 【PRINT 0】	1	整数	0	0:通常プリント2:EXPOUSEしない 1:EXPOUSEのみ行う
STSPXT	デハイス・ステータスの表示 【STSPXT】				パラメータなし
STSVIW	最大ビューポートサイズ 【STSVIW】				パラメータなし
STSZOM	拡大縮小能力の表示 【STSZOM】				パラメータなし
LUTBIT	LUTに関する能力の表示 【LUTBIT】				パラメータなし
REMOTE	リモート・ロック要求/解除 【REMOTE 1】	1	整数	1	リモート・ロック・モード 0:解除 1:要求

コマンド	機 能 【 使 用 例 】	パ ラ メ ー タ			
		No	データ型	省略値	内 容
FRPIC	フレーム単位での画像の プリント  【FRPIC 1,1,2,0】	1 2 3 4	整数 整数 整数 整数	1 1 2 0	画像サイズ番号 スタートフレーム番号 ズーム値(1:1倍 2:2倍) 0:連続,1:同期Lメモリ,2:同期Rメモリ
MOFMT	メディアのフォーマット 【MOFMT】				パラメータなし
MOINI	メディアのイニシャライズ 【MOINI 'tytle'】	1	文字列	''	メディアタイトル名
MOMID	メディアのタイトル名表示 【MOMID】				パラメータなし
MOWRT	イメージデータの書込み  【MOWRT 'file',0,0, 100,100】	1 2 3 4 5	文字列 整数 整数 整数 整数	'' — — — —	登録ファイル名 書込み領域のメモリ座標左下X座標 書込み領域のメモリ座標左下Y座標 書込み領域サイズX方向 書込み領域サイズY方向
MORED	イメージデータ表示・ ファイル名指定 【MORED 'abcd',10,10】	1 2 3	文字列 整数 整数	'' — —	表示ファイル名 表示位置のメモリ座標左下X座標 表示位置のメモリ座標左下Y座標
MORDS	イメージデータ表示・ ファイル番号指定 【MORDS 1,0,0】	1 2 3	整数 整数 整数	— — —	表示ファイル番号(1~4096) 表示位置のメモリ座標左下X座標 表示位置のメモリ座標左下Y座標
MODIR	登録ファイル名表示(ファイル名) 【MODIR 'abc'】	1	文字列	''	検索一部ファイル名
MOFIN	登録ファイル名表示(ファイル番号) 【MOFIN 1】	1	整数	—	ファイル番号(1~4096)
MODEL	登録ファイル削除 【MODEL 'abc'】	1	文字列	''	削除ファイル名
MOCMP	メディア圧縮 【MOCMP】				パラメータなし
MOFRD	イメージデータ表示 フレーム単位  【MOFRD 'file',1,1,1,0】	1 2 3 4 5	文字列 整数 整数 整数 整数	'' 1 1 1 0	登録ファイル名 画像サイズ番号 スタートフレームナンバー 登録フレーム数 メモリーチャネル選択
MOFWT	イメージデータ書込み フレーム単位 【MOFWT 'file',1,1,1,0】				MOFRDと同じ

コマンド	機 能 【 使 用 例 】	パ ラ メ ー タ			
		No	デ-タ型	省略値	内 容
RUN	shellコマンドの実行 【RUN 'ls -l'】	1	文字列	—	shellコマンド文字列

# GRAPHICA ユーティリティー説明

GRAPHICA ユーティリティーはユーザレベル(ATR)で開発したグラフィカ利用ツール類である。

これまでに開発したものを次ページ以降に示す。

尚、これらのソースは

CS05:/usr/graphica/local/src

実行モジュールは

CS05:/usr/graphica/local/bin

に格納されている。 今後新たに開発したものはここに格納し、共同利用を勧める。

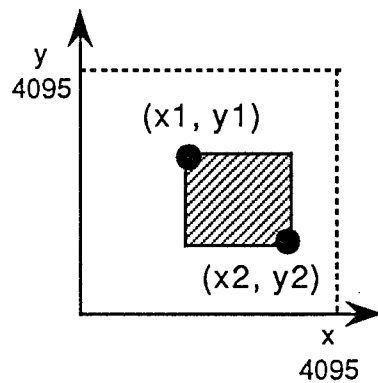
プログラム名: imgwrite

注) ソースファイル名はプログラム名.c

機能: イメージメモリ上の任意サイズの領域をホストの  
ディスクファイルに書き込む。

使用法:

- 1) 出力ファイル名の指定 (拡張子 .r,.g,.b が自動付与される)
- 2) イメージメモリ上の領域指定  $(x_1, y_1)$ ,  $(x_2, y_2)$



備考: イメージメモリのページは0に限る。



プログラム名: frmwrite

機能: イメージメモリー上の1フレームをディスクに書き込む。

使用法:

- 1) 出力ファイル名指定 (拡張子は .r,.g,.b)
- 2) フレーム番号指定 (1~512)

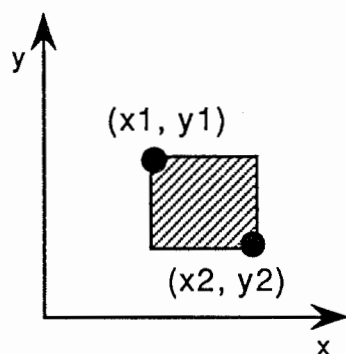
備考:

プログラム名: imgread

機能: 任意サイズの画像をディスクからイメージメモリ上に読み込む。

使用法:

- 1) 入力画像ファイル名指定
- 2) イメージメモリ上の位置指定  $(x1, y1), (x2, y2)$



備考: イメージメモリのパージは0に限る。

プログラム名: frmread

機能: ディスク上の 512 x 512 画素の画像を  
イメージメモリ上に読み込む。

使用法:

- 1) 入力画像ファイル名指定 (拡張子 .r,.g,.b)
- 2) 出力フレーム番号指定 (1~512)

備考:

プログラム名: frmredbw

機能: ディスク上の 512 x 512 x 8 bit の画像を  
イメージメモリ上に読み込む。

使用法:

- 1) 入力画像ファイル名指定
- 2) フレーム番号指定 (1~512)

備考:

プログラム名: frmreadbw4

機能: ディスクから 512 x 512 画素の画像を入力し、  
これを 256 x 256 画素の画像4枚に変換して  
グラフィカ上に表示する。

使用法:

- 1) 入力画像ファイル名指定
- 2) 出力するフレーム番号指定 (1~512)

備考:

512 x 512 x 8 bit の画像に限る。

プログラム名: frmread-s, sfrmread

機能: ディスクから 512 x 512 画素の画像を複数枚  
連続してイメージメモリ上に読み込む。

使用法:

- |                          |            |
|--------------------------|------------|
| 1) 入力画像ファイル名入力           | ファイル名 0001 |
| 2) 出力フレーム番号指定 (スタート、エンド) | ファイル名 0999 |

例えばスタート=10, エンド=20 と指定すると

ファイル名 0010 ~ ファイル名 0020 がフレーム 10 ~ 20に

読み込まれる。

備考:

プログラム名: histimg

機能: ディスク上の 512 x 512 x 8 bit の画像を  
入力し、そのヒストグラムとともにグラフィカに  
表示する。

使用法:

- 1) 入力画像ファイル名指定
- 2) 出力フレーム番号指定 (1~512)

備考:

プログラム名: histrgb

機能: ディスクから画像を入力し、R,G,B 成分の  
ヒストグラムとともにグラフィカに表示する。

使用法:

- 1) 入力画像ファイル名指定 (拡張子は .r,.g,.b)
- 2) 出力フレーム番号指定 (1~512)

備考:



プログラム名: lineimg

機能: ディスクから 512 x 512 x 8 bit の画像を入力し  
任意の水平ラインの輝度変化グラフとともに  
グラフィカに表示する。

使用法:

- 1) 入力画像ファイル名
- 2) グラフ表示したいライン番号指定 (1~512、999で終了)

備考:

プログラム名: linergb

機能: ディスク上の画像を読み込み任意の水平ライン上の  
rgb成分の輝度変化のグラフとともにグラフィカに  
表示する。

使用法:

- 1) 入力ファイル名の指定 (拡張子は .r,.g,.b)
- 2) ライン番号指定 (1～512, 999で終了)

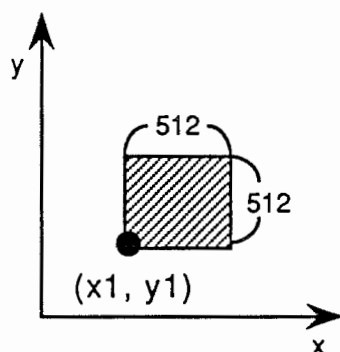
備考: 画像は、512 x 512 のサイズに限る。

プログラム名: pic512f

機能: イメージメモリ上の 512 x 512 の領域を  
ピクโตรグラフィャー2000にフルサイズで出力する。

使用法:

- 1) 領域位置の指定 (x1, y1)



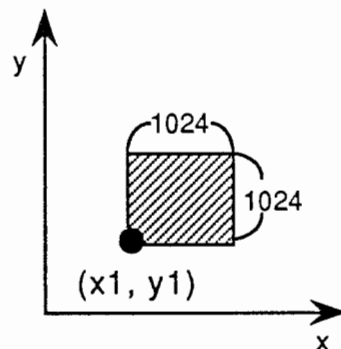
備考:

プログラム名: pic512-4

機能: イメージメモリ上の 1024 x 1024 の領域を  
ピクโตรグラフィー2000にフルサイズで出力する。

使用法:

1) 領域位置の指定 (x1, y1)



備考:

x1, y1=(0,0) と指定すればフレーム 1, 2, 9, 10

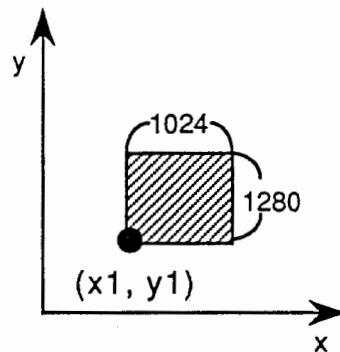
の4枚の画像が一度に出力されることになる。

プログラム名: pic1000, pic1000d

機能: イメージメモリ上の 1024 x 1280 の領域を  
ピクトグラフィャー2000に出力する。  
(1000はそのままのサイズ、1000dはフルサイズで  
出力する。)

使用法:

1) 領域の位置指定 (x1, y1)



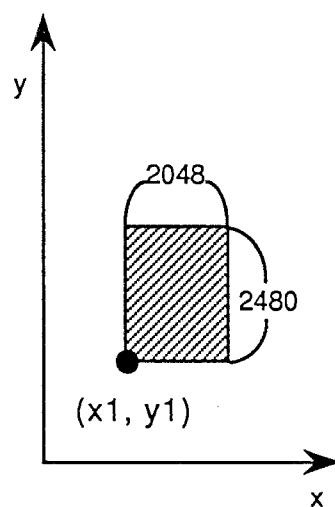
備考:

プログラム名: pic2000

機能: イメージメモリ上の任意の 2048 x 2560 画素の領域を  
ピクトログラフィー2000に出力する。

使用法:

- 1) 領域の位置指定 (x1, y1)



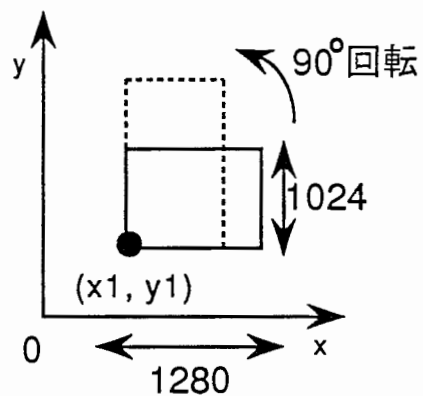
備考:

プログラム名: rot90

機能: イメージメモリー上の画像を  $90^\circ$  回転する。

使用法:

- 1) 画像の原点を指定 (x1, y1)



備考:

画像のサイズは  $1280 \times 1024$  に限る。

ピクโตรグラフィへの画像出力時に使う。

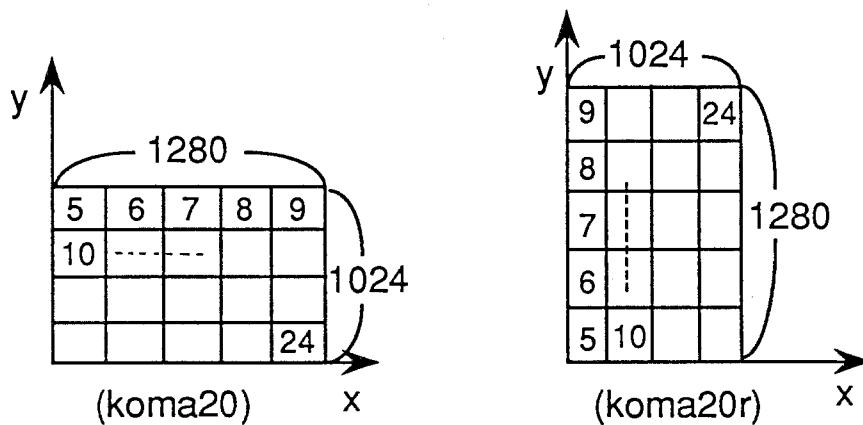
プログラム名: koma20r, koma20

機能: イメージメモリ上の連結する20フレームの画像を縮小し、20コマから成る 1280 x 1240 の画像を生成する。

使用法:

1) 開始フレーム番号の指定 (1~45)

例えば5と指定すると第5フレームから第24フレームまでが縮小される。



備考:



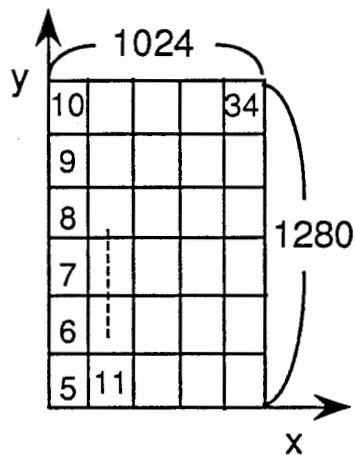
プログラム名: koma30

機能: イメージメモリ上の連続する 30フレームの画像を縮小し 30コマから成る 1280 x 1240 の画像を生成する。

使用法:

1) 開始フレーム番号の指定 (1~35)

例えば 5 を指定すると第5フレームから第34フレームまでが縮小される。



備考:

縮小時は線形補間を行なう。

プログラム名: con4000, con4000c

機能: イメージメモリを指定色でクリアする。

使用法:

- 1) クリアするレベルの指定 \*
- 2) ページの指定 (0~7, 9で全部)

\*: con4000では (0~255)

con4000cでは R,G,B 各(0~255)

備考:

プログラム名: memint

機能: R,G,B およびキャラクタープレーンをすべて  
ゼロクリアする。

使用法: 引数なし

備考:

付録:

GRAPHICAユーティリティー

ソースリスト

```

/*****
program    imgwrite.c
*****/

#include <stdio.h>
#include <math.h>
#define  Tate 2048
#define  yoko 2048
unsigned char  r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];

main()
{
    char  fname1[64],fname2[64],fname3[64];
    char  fname1r[64],fname2r[64];
    char  fname1g[64],fname2g[64];
    char  fname1b[64],fname2b[64];
    int   i,j,ta,yo;

    int   DCB[6];
    int   FMODE,SIZ,FNO,RD,RLMD,M,FUNC,STR,x1,x2,y1,y2;
    int   RB,GB,BB;
    long  NBYT;

    printf("==== N x M color image --> file write ====\n");
    printf("ENTER! RGB file name .r .g .b: ");
    scanf("%s",fname1);
    printf("\n");
    printf("UL(x1,y1),LR(x2,y2) ?:");
    scanf("%d,%d,%d,%d",&x1,&y1,&x2,&y2);
    ta = y1-y2+1;
    yo = x2-x1+1;
    printf("\n");
    sprintf(fname1r,"%s.r",fname1);
    sprintf(fname1g,"%s.g",fname1);
    sprintf(fname1b,"%s.b",fname1);

    opngd_(&DCB[0]);
    FMODE = SIZ = 1;
    RD = 2;
    RLMD = 0;
    RB=4; GB=2; BB=1;
    STR = 0;
    FUNC = 1536; /** read:1536 write:1280 **/
    FNO = 1;
    NBYT = (x2-x1+1)*(y1-y2+1);
    printf("NBYT = %d\n",NBYT);
    frmtvo_(&DCB[0],&STR,&FNO);
    dmast_(&DCB[0],&x1,&y1,&x2,&y2);
    memsl_(&DCB[0],&RB);
    imgio_(&DCB[0],&FUNC,&NBYT,&r1[0][0],&M);
    dmast_(&DCB[0],&x1,&y1,&x2,&y2);
    memsl_(&DCB[0],&GB);
    imgio_(&DCB[0],&FUNC,&NBYT,&g1[0][0],&M);
    dmast_(&DCB[0],&x1,&y1,&x2,&y2);
    memsl_(&DCB[0],&BB);
    imgio_(&DCB[0],&FUNC,&NBYT,&b1[0][0],&M);
    FNO=((int)y2/512)*8+(int)x1/512+1;
    printf("FNO = %d\n",FNO);
    STR = 1;
    frmtvo_(&DCB[0],&STR,&FNO);
    clsgd_(&DCB[0]);

    writeimg(r1,ta,yo,fname1r);
    writeimg(g1,ta,yo,fname1g);
    writeimg(b1,ta,yo,fname1b);
}

```

```

}

/*****/
reading(buff,t_size,y_size,f_name)
/*****/
unsigned char buff[];
int      t_size,y_size;
char     f_name[];
{
    FILE      *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(buff,t_size,y_size,f_pointer);
    fclose (f_pointer);
}

/*****/
writing(buff,t_size,y_size,f_name)
/*****/
unsigned char buff[];
int      t_size,y_size;
char     f_name[];
{
    FILE      *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"w")) == NULL)
        { printf("\nfile create error : %s",f_name);
          exit(); }
    printf("writing... %s\n",f_name);
    fwrite(buff,t_size,y_size,f_pointer);
    fclose (f_pointer);
}

/*****/
program    limgb.c
*****/

#include <stdio.h>
#include <math.h>
#define  tate  512
#define  yoko  512
#define  thp   128 /** BW hanten **/
unsigned char  r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];
unsigned char  r2[tate][yoko],g2[tate][yoko],b2[tate][yoko];

main()
{
    char  fname1[64],fname2[64],fname3[64];
    char  fname1r[64],fname2r[64];
    char  fname1g[64],fname2g[64];
    char  fname1b[64],fname2b[64];
    int   i,j;
    int   lin;
    int   DCB[6];
    int   FMOD,SIZ,FNO,RD,RLMD,STN;

    printf("=== 512x512 image & scanline -> graphica ===\n");
    printf("ENTER!  file name .r .g .b: ");
    scanf("%s",fname1);
    sprintf(fname1r,"%s.r",fname1);
    sprintf(fname1g,"%s.g",fname1);
    sprintf(fname1b,"%s.b",fname1);
    reading(r1,tate,yoko,fname1r);
    reading(g1,tate,yoko,fname1g);
    reading(b1,tate,yoko,fname1b);
}

```

```

printf("ENTER! line number (1-512,END=999): ");
scanf("%d",&lin);
while (lin != 999) {
    printf("FRAME NUMBER? :");
    scanf("%d",&FNO);
    printf("\n");
    printf("pre-process...\n");
    for (j=0;j<tate;j++) {
        for (i=0;i<yoko;i++) {
            r2[j][i] = r1[j][i];
            g2[j][i] = g1[j][i];
            b2[j][i] = b1[j][i];
        }
    }

    printf("linescan.....\n");
    line(lin);

    opngd_(&DCB[0]);
    FMOD = SIZ = STN=1;
    STN = 0;
    frmtvo_(&DCB[0],&STN,&FNO);
    RD = 2;
    RLMD = 0;
    frmio_(&DCB[0],&FMOD,&SIZ,&FNO,&r2[0][0],&g2[0][0],&b2[0][0],&RD,&RLMD);
    STN = 1;
    frmtvo_(&DCB[0],&STN,&FNO);
    clsgd_(&DCB[0]);

    printf("ENTER! line number (1-512,END=999): ");
    scanf("%d",&lin);
}

```

}

/\*\*\*\*\*\*

line(lnum)

/\*\*\*\*\*\*

int lnum;

{

int i,j,k1,k2;

int bbb=50;

int on=255;

int off=0;

```

for (i=0;i<yoko;i++) {
    k1 = tate - (int)b1[lnum][i] - bbb;
    if (i<yoko-2) {
        k2 = tate - (int)b1[lnum][i+1] - bbb;
        plot2(i,k1,k2,off,off,on);
    }
    else plot(i,k1,off,off,on);
}

```

```

for (i=0;i<yoko;i++) {
    k1 = tate - (int)g1[lnum][i] - bbb;
    if (i<yoko-2) {
        k2 = tate - (int)g1[lnum][i+1] - bbb;
        plot2(i,k1,k2,off,on,off);
    }
    else plot(i,k1,off,on,off);
}

```

```

for (i=0;i<yoko;i++) {
    k1 = tate - (int)r1[lnum][i] - bbb;
    if (i<yoko-2) {

```

```

        k2 = tate - (int)r1[lnum][i+1] - bbb;
        plot2(i,k1,k2,on,off,off);
    }
    else plot(i,k1,on,off,off);
    plot(i,lnum,on,on,on);
}

/*****/
plot(x,y,r,g,b)
/*****/
int x,y,r,g,b;
{
    r2[y][x] = r;
    g2[y][x] = g;
    b2[y][x] = b;
}

/*****/
plot2(x0,y1,y2,r,g,b)
/*****/
int x0,y1,y2,r,g,b;
{
    int i,xx,yy,yh,yd;

    yd = abs(y1-y2);
    yh = yd/2;
    for (i=0;i<=yd;i++) {
        if (y1<y2) yy=y1+i;
        if (y1>=y2) yy=y1-i;
        if (i<=yd) xx=x0;
        if (i>yd) xx=x0+1;
        r2[yy][xx] = r;
        g2[yy][xx] = g;
        b2[yy][xx] = b;
    }
}

/*****/
reading(buff,t_size,y_size,f_name)
/*****/
unsigned char buff[];
int t_size,y_size;
char f_name[];
{
    FILE *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(buff,t_size,y_size,f_pointer);
    fclose (f_pointer);
}

/*****/
program rot90.c
/*****/

#include <stdio.h>
#include <math.h>
#define tate 1024
#define yoko 1280
unsigned char m1[tate][yoko],m2[yoko][tate];

```



```

main()
{
    int i,j;

    int DCB[6];
    int FMODE,SIZ,FNO,RD,RLMD,M,FUNCR,FUNCW,STR,x1,x2,y1,y2;
    int xx1,xx2,yy1,yy2,xxd,yyd,x0,y0;
    int RB,GB,BB;
    long NBYT;

    printf("==== ROTAE 90deg image on graphica ====\\n");
    printf("==== image size 1280 x 1024 ====\\n");
    printf("org of image LL(x0,y0)?");
    scanf("%d,%d",&x0,&y0);
    printf("\\n");
    x1=x0; y1=y0+tate-1; x2=x0+yoko-1; y2=y0;
    xx1=x1; xx2=y1; yy1=x2; yy2=y2;
    xxd=x2-x1+1; yyd=y1-y2+1;
    printf("(\\d,\\d)-(\\d,\\d) --> (\\d,\\d)-(\\d,\\d)\\n",
        x1,y1,x2,y2,xx1,yy1,xx2,yy2);

    opngd_(&DCB[0]);
    FMODE = SIZ = 1;
    RD = 2;
    RLMD = 0;
    RB=4; GB=2; BB=1;
    STR = 0;
    FUNCW = 1280; FUNCR = 1536;
    FNO = 1;
    NBYT = (x2-x1+1)*(y1-y2+1);

    frmtvo_(&DCB[0],&STR,&FNO);

    dmast_(&DCB[0],&x1,&y1,&x2,&y2);
    memsl_(&DCB[0],&RB);
    imgio_(&DCB[0],&FUNCR,&NBYT,&m1[0][0],&M);
    rotate(xxd,yyd);
    dmast_(&DCB[0],&xx1,&yy1,&xx2,&yy2);
    memsl_(&DCB[0],&RB);
    imgio_(&DCB[0],&FUNCW,&NBYT,&m2[0][0],&M);

    dmast_(&DCB[0],&x1,&y1,&x2,&y2);
    memsl_(&DCB[0],&GB);
    imgio_(&DCB[0],&FUNCR,&NBYT,&m1[0][0],&M);
    rotate(xxd,yyd);
    dmast_(&DCB[0],&xx1,&yy1,&xx2,&yy2);
    memsl_(&DCB[0],&GB);
    imgio_(&DCB[0],&FUNCW,&NBYT,&m2[0][0],&M);

    dmast_(&DCB[0],&x1,&y1,&x2,&y2);
    memsl_(&DCB[0],&BB);
    imgio_(&DCB[0],&FUNCR,&NBYT,&m1[0][0],&M);
    rotate(xxd,yyd);
    dmast_(&DCB[0],&xx1,&yy1,&xx2,&yy2);
    memsl_(&DCB[0],&BB);
    imgio_(&DCB[0],&FUNCW,&NBYT,&m2[0][0],&M);

    FNO=((int)yy2/512)*8+(int)xx1/512+1;
    printf("FNO = \\d\\n",FNO);
    STR = 1;
    frmtvo_(&DCB[0],&STR,&FNO);
    clsgd_(&DCB[0]);
}

```

```

/*****/

```

```
rotate(xd,yd)
/******/
int  xd,yd;
{
    int  i,j,i2,j2;

    for(j=0;j<yd;j++) {
        j2=yd-j-1;
        for (i=0;i<xd;i++) {
            i2=xd-i-1;
            m2[i2][j] = m1[j][i];
        }
    }
}
```

~~fclose(f\_pointer);~~

~~\*/~~

```
/*
*****
program   frmwrite.c
*****/

#include <stdio.h>
#include <math.h>
#define  tate 512
#define  yoko 512
unsigned char  r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];

main()
{
    char  fname1[64],fname2[64],fname3[64];
    char  fname1r[64],fname2r[64];
    char  fname1g[64],fname2g[64];
    char  fname1b[64],fname2b[64];
    int   i,j;

    int   DCB[6];
    int   FMOD,SIZ,FNO,RD,RLMD,STN;

    printf("=== graphica -> 512x512 color image ===\n");
    printf("ENTER! RGB file name .r .g .b: ");
    scanf("%s",fname1);
    printf("\n");
    printf("FRAME NUMBER? :");
    scanf("%d",&FNO);
    sprintf(fname1r,"%s.r",fname1);
    sprintf(fname1g,"%s.g",fname1);
    sprintf(fname1b,"%s.b",fname1);
    opngd_(&DCB[0]);
    FMOD = SIZ = STN=1;
    STN = 0;
    STN = 1;
    RD = 1;
    RLMD = 0;
    frmio_(&DCB[0],&FMOD,&SIZ,&FNO,&r1[0][0],&g1[0][0],&b1[0][0],&RD,&RLMD);
    STN = 1;
    frmtvo_(&DCB[0],&STN,&FNO);
    clsgd_(&DCB[0]);

    for (j=480;j<tate;j++)
        for(i=0;i<yoko;i++) {
            r1[tate][yoko]=255;
            g1[tate][yoko]=255;
            b1[tate][yoko]=255;
        }
    writeimg(r1,tate,yoko,fname1r);
    writeimg(g1,tate,yoko,fname1g);
    writeimg(b1,tate,yoko,fname1b);
}

/*
*****/
reading(buff,t_size,y_size,f_name)
/*
*****/
unsigned char buff[];
int          t_size,y_size;
char         f_name[];
{
    FILE          *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);

```

```

        exit(); }
    printf("reading... %s\n",f_name);
    fread(buff,t_size,y_size,f_pointer);
    fclose (f_pointer);
}
/*****/
writeimg(buff,t_size,y_size,f_name)
/*****/
unsigned char buff[];
int      t_size,y_size;
char     f_name[];
{
    FILE      *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"w")) == NULL)
        { printf("\nfile create error : %s",f_name);
          exit(); }
    printf("writing... %s\n",f_name);
    fwrite(buff,t_size,y_size,f_pointer);
    fclose (f_pointer);
}
/*****
program   imgread.c
*****/

#include <stdio.h>
#include <math.h>
#define  Tate 2048
#define  yoko 2048
unsigned char  r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];

main()
{
    char  fname1[64],fname2[64],fname3[64];
    char  fname1r[64],fname2r[64];
    char  fname1g[64],fname2g[64];
    char  fname1b[64],fname2b[64];
    int   i,j;

    int   DCB[6];
    int   FMOD,SIZ,FNO,RD,RLMD,M,FUNC,STR,x1,x2,y1,y2;
    int   RB,GB,BB;
    long  NBYT;

    printf("==== N x M color image --> graphica ==== \n");
    printf("ENTER! RGB file name .r .g .b: ");
    scanf("%s",fname1);
    printf("\n");
    printf("UL(x1,y1),LR(x2,y2) ? :");
    scanf("%d,%d,%d,%d",&x1,&y1,&x2,&y2);
    printf("\n");
    sprintf(fname1r,"%s.r",fname1);
    sprintf(fname1g,"%s.g",fname1);
    sprintf(fname1b,"%s.b",fname1);
    readimg_r(fname1r);
    readimg_g(fname1g);
    readimg_b(fname1b);

    opngd_(&DCB[0]);
    FMOD = SIZ = 1;
    RD = 2;
    RLMD = 0;
    RB=4; GB=2; BB=1;
    STR = 0;
    FUNC = 1280;
    FNO = 1;

```

```

    NBYT = (x2-x1+1)*(y1-y2+1);
    printf("NBYT = %d\n",NBYT);
    frmtvo_(&DCB[0],&STR,&FNO);
    dmast_(&DCB[0],&x1,&y1,&x2,&y2);
    memsl_(&DCB[0],&RB);
    imgio_(&DCB[0],&FUNC,&NBYT,&r1[0][0],&M);
    dmast_(&DCB[0],&x1,&y1,&x2,&y2);
    memsl_(&DCB[0],&GB);
    imgio_(&DCB[0],&FUNC,&NBYT,&g1[0][0],&M);
    dmast_(&DCB[0],&x1,&y1,&x2,&y2);
    memsl_(&DCB[0],&BB);
    imgio_(&DCB[0],&FUNC,&NBYT,&b1[0][0],&M);
    FNO=((int)y2/512)*8+(int)x1/512+1;
    printf("FNO = %d\n",FNO);
    STR = 1;
    frmtvo_(&DCB[0],&STR,&FNO);
    clsgd_(&DCB[0]);
}

/*****/
readingr(f_name)
/*****/
char      f_name[];
{
    FILE      *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(r1,tate,yoko,f_pointer);
    fclose (f_pointer);
}

/*****/
readingg(f_name)
/*****/
char      f_name[];
{
    FILE      *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(g1,tate,yoko,f_pointer);
    fclose (f_pointer);
}

/*****/
readingb(f_name)
/*****/
char      f_name[];
{
    FILE      *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(b1,tate,yoko,f_pointer);
    fclose (f_pointer);
}

/*****/
program    lineimg.c
/*****/

```

```

#include <stdio.h>
#include <math.h>
#define tate 512
#define yoko 512
#define thp 128 /** BW hanten **/
unsigned char r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];

main()
{
    char fname1[64];
    int i,j;
    int lin;
    int DCB[6];
    int FMOD,SIZ,FNO,RD,RLMD,STN;

    printf("=== 512x512 image & scanline -> graphica ===\n");
    printf("ENTER! file name : ");
    scanf("%s",fname1);
    reading(r1,tate,yoko,fname1);
    printf("ENTER! line number (1-512,END=999): ");
    scanf("%d",&lin);
    while (lin != 999) {
        printf("FRAME NUMBER? :");
        scanf("%d",&FNO);
        printf("\n");
        printf("pre-process...\n");
        for (j=0;j<tate;j++) {
            for (i=0;i<yoko;i++) {
                g1[j][i] = r1[j][i];
            }
        }

        printf("linescan.....\n");
        line(lin);

        opngd_(&DCB[0]);
        FMOD = SIZ = STN=1;
        STN = 0;
        frmtvo_(&DCB[0],&STN,&FNO);
        RD = 2;
        RLMD = 0;
        frmio_(&DCB[0],&FMOD,&SIZ,&FNO,&g1[0][0],&g1[0][0],&g1[0][0],&RD,&RLMD);
        STN = 1;
        frmtvo_(&DCB[0],&STN,&FNO);
        clsgd_(&DCB[0]);

        printf("ENTER! line number (1-512,END=999): ");
        scanf("%d",&lin);
    }
}

/*****/
line(lnum)
/*****/
int lnum;
{
    int i,j,k1,k2;
    int bbb=50;

    for (i=0;i<yoko;i=i++) {
        k1 = tate - (int)r1[lnum][i] - bbb;
        if (i<yoko-2) {
            k2 = tate - (int)r1[lnum][i+1] - bbb;
            plot2(i,k1,k2);
        }
    }
}

```

```

        else plot(i,k1);
        plot(i,lnum);
    }
}

/*****/
plot(x,y)
/*****/
int x,y;
{
    int wht=255;
    int blk=0;

    if (r1[y][x]<=thp) {
        g1[y][x] = wht;
    }
    else {
        g1[y][x] = blk;
    }
}

/*****/
plot2(x0,y1,y2)
/*****/
int x0,y1,y2;
{
    int i,xx,yy,yh,yd;
    int wht=255;
    int blk=0;

    yd = abs(y1-y2);
    yh = yd/2;
    for (i=0;i<=yd;i++) {
        if (y1<y2) yy=y1+i;
        if (y1>=y2) yy=y1-i;
        if (i<=yd) xx=x0;
        if (i>yd) xx=x0+1;
        if (r1[yy][xx]<=thp) {
            g1[yy][xx] = wht;
        }
        else {
            g1[yy][xx] = blk;
        }
    }
}

/*****/
reading(buff,t_size,y_size,f_name)
/*****/
unsigned char buff[];
int t_size,y_size;
char f_name[];
{
    FILE *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(buff,t_size,y_size,f_pointer);
    fclose (f_pointer);
}

#include <stdio.h>
#include <math.h>

```

```

int     DCB[6],BUSY,LOCK,PRNT,ERR[3],ID,REM;
int     WIND[4],VIEW[4],READ,W[4],V[4];
int     EXPOS;
int     STR,TFN;
int     i;
main()
{
    for (i=0;i<6;i++) DCB[i]=0;
    opngd_(&DCB[0]);
    opnpct_(&DCB[0]);
    REM = 1; LOCK=0;
    remote_(&DCB[0],&REM);
    while(LOCK==0) {
        stspxt_(&DCB[0],&BUSY,&LOCK,&PRNT,&ERR[0],&ID);
        printf("BUSY=%d LOCK=%d PRNT=%d ERR:%d %d %d ID=%d\n",
            BUSY,LOCK,PRNT,ERR[0],ERR[1],ERR[2],ID);
    }
    READ = 1;
    wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        WIND[0],WIND[1],WIND[2],WIND[3],
        VIEW[0],VIEW[1],VIEW[2],VIEW[3]);

    /*** window-viewport settei ****/
    printf("ENTER ! WIND(x1,y1)(x2,y2) VIEW(x1,y1)(x2,y2):");
    scanf("%d,%d,%d,%d,%d,%d,%d,%d",&W[0],&W[1],&W[2],&W[3],&V[0],&V[1],&V[2],&V[3]);
    printf("\n");
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        W[0],W[1],W[2],W[3],
        V[0],V[1],V[2],V[3]);

    READ = 0;
    wndviw_(&DCB[0],&READ,&W[0],&V[0]);
    READ = 1;
    wndviw_(&DCB[0],&READ,&W[0],&V[0]);
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        W[0],W[1],W[2],W[3],
        V[0],V[1],V[2],V[3]);

    STR = 0; TFN = 1;
    frmtvo_(&DCB[0],&STR,&TFN);
    EXPOS = 0;
    print_(&DCB[0],&EXPOS);

    /*** moto no settei ni modosu ****/
    READ = 0;
    wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
    READ = 1;
    wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        WIND[0],WIND[1],WIND[2],WIND[3],
        VIEW[0],VIEW[1],VIEW[2],VIEW[3]);

    REM = 0;
    remote_(&DCB[0],&REM);
    stspxt_(&DCB[0],&BUSY,&LOCK,&PRNT,&ERR[0],&ID);
    printf("BUSY=%d LOCK=%d PRNT=%d ERR:%d %d %d ID=%d\n",
        BUSY,LOCK,PRNT,ERR[0],ERR[1],ERR[2],ID);

    clsgd_(&DCB[0]);
}

```



```

/*****
program   frmreadbw4.c
*****/

#include <stdio.h>
#include <math.h>
#define   tate   512
#define   yoko   512
unsigned char  r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];
unsigned char  r2[tate][yoko],g2[tate][yoko],b2[tate][yoko];

main()
{
    char  fname1[64];
    int   i,j;

    int   DCB[6];
    int   FMOD,SIZ,FNO,RD,RLMD,STN;

    printf("== 512x512 B/W image -> 1/4 size x 4 -> graphica ==\n");
    printf("ENTER! B/W file name ");
    scanf("%s",fname1);
    printf("\n");
    printf("FRAME NUMBER? :");
    scanf("%d",&FNO);
    readingr(fname1);
    readingg(fname1);
    readingb(fname1);

    reduce4();

    opngd_(&DCB[0]);
    FMOD = SIZ = STN=1;
    STN = 0;
    frmtvo_(&DCB[0],&STN,&FNO);
    RD = 2;
    RLMD = 0;
    frmio_(&DCB[0],&FMOD,&SIZ,&FNO,&r2[0][0],&g2[0][0],&b2[0][0],&RD,&RLMD);
    STN = 1;
    frmtvo_(&DCB[0],&STN,&FNO);
    clsgd_(&DCB[0]);
}

/*****/
readingr(f_name)
/*****/
char      f_name[];
{
    FILE      *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(r1,tate,yoko,f_pointer);
    fclose (f_pointer);
}

/*****/
readingg(f_name)
/*****/
char      f_name[];
{
    FILE      *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)

```

```

        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(g1,tate,yoko,f_pointer);
    fclose (f_pointer);
}
/*****/
readingb(f_name)
/*****/
char    f_name[];
{
    FILE    *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(b1,tate,yoko,f_pointer);
    fclose (f_pointer);
}

/*****/
reduce4()
/*****/
{
    int    i,j,im;

    for (j=0;j<tate/2;j++) {
        for (i=0;i<yoko/2;i++) {
            im = (int)((float)r1[j*2][i*2] +
                      (float)r1[j*2+1][i*2] +
                      (float)r1[j*2][i*2+1] +
                      (float)r1[j*2+1][i*2+1] ) /4.0);
            r2[j][i] = r2[j+256][i] = r2 [j][i+256] = r2[j+256][i+256] =
            g2[j][i] = g2[j+256][i] = g2 [j][i+256] = g2[j+256][i+256] =
            b2[j][i] = b2[j+256][i] = b2 [j][i+256] = b2[j+256][i+256] =
            im;
        }
    }
}

/*****/
program    histrgb.c
/*****/

#include <stdio.h>
#include <math.h>
#define    tate    480
#define    yoko    512
#define    thp    128 /** BW hanten **/
unsigned char    r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];
long    hr[256],hg[256],hb[256];
int    hrr[256],hgg[256],hbb[256];

main()
{
    char    fname1[64],fname2[64],fname3[64];
    char    fname1r[64],fname2r[64];
    char    fname1g[64],fname2g[64];
    char    fname1b[64],fname2b[64];
    int    i,j;
    int    lin;
    int    DCB[6];
    int    FMOD,SIZ,FNO,RD,RLMD,STN;

    printf("=== 512x480 image & histogram -> graphica ===\n");
}

```

```

printf("ENTER! file name .r .g .b: ");
scanf("%s", fname1);
printf("ENTER! FRAME NUMBER: ");
scanf("%d", &FNO);
printf("\n");
sprintf(fname1r, "%s.r", fname1);
sprintf(fname1g, "%s.g", fname1);
sprintf(fname1b, "%s.b", fname1);
reading(r1, tate, yoko, fname1r);
reading(g1, tate, yoko, fname1g);
reading(b1, tate, yoko, fname1b);
for (i=0; i<256; i++) {
    hr[i] = hg[i] = hb[i] = 0;
}

printf("histgram.....\n");
hist();

opngd_(&DCB[0]);
FMOD = SIZ = STN=1;
STN = 0;
frmtvo_(&DCB[0], &STN, &FNO);
RD = 2;
RLMD = 0;
frmio_(&DCB[0], &FMOD, &SIZ, &FNO, &r1[0][0], &g1[0][0], &b1[0][0], &RD, &RLMD);
STN = 1;
frmtvo_(&DCB[0], &STN, &FNO);
clsgd_(&DCB[0]);
}

/*****/
hist()
/*****/
{
    int i, j, k1, k2, lnum;
    int i1, i2, i3;
    int bbb=50;
    int on=255;
    int off=0;
    int hight=300;
    long mmm=0;

    for (j=0; j<tate; j=j++) {
        for (i=0; i<yoko; i=i++) {
            i1=r1[j][i]; hr[i1]=hr[i1]+1;
            i2=g1[j][i]; hg[i2]=hg[i2]+1;
            i3=b1[j][i]; hb[i3]=hb[i3]+1;
        }
    }

    for (i=0; i<256; i=i++) {
        if(mmm<hr[i]) mmm=hr[i];
        if(mmm<hg[i]) mmm=hg[i];
        if(mmm<hb[i]) mmm=hb[i];
    }

    for (i=0; i<256; i=i++) {
        hrr[i*2] = hrr[i*2+1] = hight*hr[i]/mmm;
        hgg[i*2] = hgg[i*2+1] = hight*hg[i]/mmm;
        hbb[i*2] = hbb[i*2+1] = hight*hb[i]/mmm;
    }

    for (i=0; i<yoko-1; i=i++) {
        k1 = tate - (int)hbb[i] - bbb;
        k2 = tate - (int)hbb[i+1] - bbb;

```

```

    plot2(i,k1,k2,off,off,on);
}

for (i=0;i<yoko-1;i=i++) {
    k1 = tate - (int)hgg[i] - bbb;
    k2 = tate - (int)hgg[i+1] - bbb;
    plot2(i,k1,k2,off,on,off);
}

for (i=0;i<yoko-1;i=i++) {
    k1 = tate - (int)hrr[i] - bbb;
    k2 = tate - (int)hrr[i+1] - bbb;
    plot2(i,k1,k2,on,off,off);
    lnum = 512 - bbb;
    plot(i,lnum,on,on,on);
}
}

/*****/
plot(x,y,r,g,b)
/*****/
int x,y,r,g,b;
{

    r1[y][x] = r;
    g1[y][x] = g;
    b1[y][x] = b;
}

/*****/
plot2(x0,y1,y2,r,g,b)
/*****/
int x0,y1,y2,r,g,b;
{
    int i,xx,yy,yh,yd;

    yd = abs(y1-y2);
    yh = yd/2;
    for (i=0;i<=yd;i++) {
        if (y1<y2) yy=y1+i;
        if (y1>=y2) yy=y1-i;
        if (i<=yd) xx=x0;
        if (i>yd) xx=x0+1;
        r1[yy][xx] = r;
        g1[yy][xx] = g;
        b1[yy][xx] = b;
    }
}

/*****/
reading(buff,t_size,y_size,f_name)
/*****/
unsigned char buff[];
int t_size,y_size;
char f_name[];
{
    FILE *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(buff,t_size,y_size,f_pointer);
    fclose (f_pointer);
}

```

```

    b=fm;
}
if((hd >= 1.) && (hd < 2.)) {
    r=fn;
    g=vnrn;
    b=fm;
}
if((hd >= 2.) && (hd < 3.)) {
    r=fm;
    g=vnrn;
    b=fk;
}
if((hd >= 3.) && (hd < 4.)) {
    r=fm;
    g=fn;
    b=vnrn;
}
if((hd >= 4.) && (hd < 5.)) {
    r=fk;
    g=fn;
    b=vnrn;
}
if(hd >= 5.) {
    r=vnrn;
    g=fn;
    b=fn;
}

    ir = r * 255.;
    ig = g * 255.;
    ib = b * 255.;
}

```

PIC512f.c

```

#include <stdio.h>
#include <math.h>

#define haba_x 511
#define haba_y 511
int    DCB[6], BUSY, LOCK, PRNT, ERR[3], ID, REM;
int    WIND[4], VIEW[4], READ, W[4], V[4], SIZ[2];
int    EXPOS;
int    STR, TFN;
int    i;
int    x, y;
main()
{
    for (i=0; i<6; i++) DCB[i]=0;
    opngd_(&DCB[0]);
    opnpct_(&DCB[0]);
    REM = 1; LOCK=0;
    printf("==== graphica 512x512 --> PICTORO 2000 =====\n");
    remote_(&DCB[0], &REM);
    while(LOCK==0) {
        stspxt_(&DCB[0], &BUSY, &LOCK, &PRNT, &ERR[0], &ID);
        printf("BUSY=%d LOCK=%d PRNT=%d ERR:%d %d %d ID=%d\n",
            BUSY, LOCK, PRNT, ERR[0], ERR[1], ERR[2], ID);
    }
    READ = 1;
    wndviw_(&DCB[0], &READ, &WIND[0], &VIEW[0]);
/**
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        WIND[0], WIND[1], WIND[2], WIND[3],
        VIEW[0], VIEW[1], VIEW[2], VIEW[3]);
**/
}

```

```

/**** window-viewport settei ****/
printf("ENTER ! ORG LL(x,y)");
scanf("%d,%d",&x,&y);
printf("\n");
W[0]=x; W[1]=y; W[2]=x+haba_x; W[3]=y+haba_y;
V[0]=0; V[1]=0; V[2]=2047; V[3]=2047;
/****
printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
      W[0],W[1],W[2],W[3],
      V[0],V[1],V[2],V[3]);
**/

READ = 0;
SIZ[0]=4;SIZ[1]=1;magsiz_(&DCB[0],&READ,&SIZ[0]);
wndviw_(&DCB[0],&READ,&W[0],&V[0]);
READ = 1;
magsiz_(&DCB[0],&READ,&SIZ[0]);
wndviw_(&DCB[0],&READ,&W[0],&V[0]);
printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
      W[0],W[1],W[2],W[3],
      V[0],V[1],V[2],V[3]);
printf("SIZE:%d %d\n",SIZ[0],SIZ[1]);

STR = 0; TFN = 1;
frmtvo_(&DCB[0],&STR,&TFN);
EXPOS = 0;
print_(&DCB[0],&EXPOS);

/**** moto no settei ni modosu ****/
READ = 0;
SIZ[0]=1;SIZ[1]=1;magsiz_(&DCB[0],&READ,&SIZ[0]);
wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
READ = 1;
magsiz_(&DCB[0],&READ,&SIZ[0]);
wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
      WIND[0],WIND[1],WIND[2],WIND[3],
      VIEW[0],VIEW[1],VIEW[2],VIEW[3]);
printf("SIZE:%d %d\n",SIZ[0],SIZ[1]);

REM = 0;
remote_(&DCB[0],&REM);
stspxt_(&DCB[0],&BUSY,&LOCK,&PRNT,&ERR[0],&ID);
printf("BUSY=%d LOCK=%d PRNT=%d ERR:%d %d %d ID=%d\n",
      BUSY,LOCK,PRNT,ERR[0],ERR[1],ERR[2],ID);

clsgd_(&DCB[0]);
}

```

```

/*****
program   frmreadbw.c
*****/

#include <stdio.h>
#include <math.h>
#define   tate  512
#define   yoko  512
unsigned char  r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];

main()
{
    char  fname1[64];
    int   i,j;

    int   DCB[6];
    int   FMODE,SIZ,FNO,RD,RLMD,STN;

    printf("=== 512x512 B/W image -> graphica ===\n");
    printf("ENTER! B/W file name ");
    scanf("%s",fname1);
    printf("\n");
    printf("FRAME NUMBER? :");
    scanf("%d",&FNO);
    readimgr(fname1);
    readimgg(fname1);
    readimgb(fname1);

    opngd_(&DCB[0]);
    FMODE = SIZ = STN=1;
    STN = 0;
    frmtvo_(&DCB[0],&STN,&FNO);
    RD = 2;
    RLMD = 0;
    frmio_(&DCB[0],&FMODE,&SIZ,&FNO,&r1[0][0],&g1[0][0],&b1[0][0],&RD,&RLMD);
    STN = 1;
    frmtvo_(&DCB[0],&STN,&FNO);
    clsgd_(&DCB[0]);
}

/*****/
readimgr(f_name)
/*****/
char      f_name[];
{
    FILE      *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(r1,tate,yoko,f_pointer);
    fclose (f_pointer);
}

/*****/
readimgg(f_name)
/*****/
char      f_name[];
{
    FILE      *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
}

```

```

        fread(g1,tate,yoko,f_pointer);
        fclose (f_pointer);
    }
    /*****/
    readingb(f_name)
    /*****/
    char        f_name[];
    {
        FILE        *f_pointer,*fopen();

        if ((f_pointer = fopen(f_name,"r")) == NULL)
            { printf("\nfile read error : %s",f_name);
              exit(); }
        printf("reading... %s\n",f_name);
        fread(b1,tate,yoko,f_pointer);
        fclose (f_pointer);
    }
    /*****
    program    histimg.c
    *****/

#include <stdio.h>
#include <math.h>
#define  tate  512
#define  yoko  512
#define  thp   128 /** BW hanten **/
#define  bbb   50
unsigned char  r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];
unsigned char  h1[tate][yoko],v1[tate][yoko],d1[tate][yoko];
long          hr[256],hg[256],hb[256];
int           hrr[256],hgg[256],hbb[256];

int           ir,ig,ib,ih,iv,id;

main()
{
    char  fname1[64],fname2[64],fname3[64];
    char  fname1r[64],fname1h[64];
    char  fname1g[64],fname1v[64];
    char  fname1b[64],fname1d[64];
    int   i,j;
    int   lin;
    int   DCB[6];
    int   FMOD,SIZ,FNO,RD,RLMD,STN;

    printf("=== 512x512 image & histogram -> graphica ===\n");
    printf("ENTER! B/W file name : ");
    scanf("%s",fname1);
    printf("ENTER! FRAME NUMBER: ");
    scanf("%d",&FNO);
    printf("\n");
    reading(r1,tate,yoko,fname1);
    reading(g1,tate,yoko,fname1);
    reading(b1,tate,yoko,fname1);
    for (i=0;i<256;i++) {
        hr[i] = 0;
    }

    printf("histogram.....\n");
    hist();

    opngd_(&DCB[0]);
    FMOD = SIZ = STN=1;
    STN = 0;
    frmtvo_(&DCB[0],&STN,&FNO);
    RD = 2;

```



```

RLMD = 0;
frmio_(&DCB[0], &FMOD, &SIZ, &FNO, &r1[0][0], &g1[0][0], &b1[0][0], &RD, &RLMD);
STN = 1;
frmtvo_(&DCB[0], &STN, &FNO);
clsgd_(&DCB[0]);

```

```

}

```

```

/*****/

```

```

hist()

```

```

/*****/

```

```

{
    int i,j,k1,k2,lnum;
    int i1,i2,i3;
    int on=255;
    int off=0;
    int hight=300;
    long mmm=0;

    for (j=0;j<tate;j=j++) {
        for (i=0;i<yoko;i=i++) {
            i1=r1[j][i]; hr[i1]=hr[i1]+1;
        }
    }

    for (i=0;i<256;i=i++) {
        if(mmm<hr[i]) mmm=hr[i];
    }

    for (i=0;i<256;i=i++) {
        hrr[i*2] = hrr[i*2+1] = hight*hr[i]/mmm;
    }
    lnum = 512 - bbb;
    for (i=0;i<yoko-1;i=i++) {
        plot(i,lnum,on,on,on);
        k1 = tate - (int)hrr[i] - bbb;
        k2 = tate - (int)hrr[i+1] - bbb;
        plot2(i,k1,k2,on,on,off);
    }
}

```

```

/*****/

```

```

plot(x,y,r,g,b)

```

```

/*****/

```

```

int x,y,r,g,b;
{
    r1[y][x] = r;
    g1[y][x] = g;
    b1[y][x] = b;
}

```

```

/*****/

```

```

plot2(x0,y1,y2,r,g,b)

```

```

/*****/

```

```

int x0,y1,y2,r,g,b;
{
    int i,xx,yy,yh,yd;

    yd = abs(y1-y2);
    yh = yd/2;
    for (i=0;i<=yd;i++) {
        if (y1<y2) yy=y1+i;
        if (y1>=y2) yy=y1-i;
        if (i<=yd) xx=x0;
    }
}

```

```

        if (i>yd) xx=x0+1;
        r1[yy][xx] = r;
        g1[yy][xx] = g;
        b1[yy][xx] = b;
    }
}

/*****/
plot2h(x0,y1,y2,r,g,b)
/*****/
int x0,y1,y2,r,g,b;
{
    int i,xx,yy,yh,yd;

    yd = abs(y1-y2);
    yh = yd/2;

    ih = x0/2;
    iv = 250; id = 250;
    hvdtoRGB();

    for (i=0;i<=yd;i++) {
        if (y1<y2) yy=y1+i;
        if (y1>=y2) yy=y1-i;
        if (i<=yd) xx=x0;
        if (i>yd) xx=x0+1;
        r1[yy][xx] = ir;
        g1[yy][xx] = ig;
        b1[yy][xx] = ib;
    }
}

/*****/
reading(buff,t_size,y_size,f_name)
/*****/
unsigned char buff[];
int t_size,y_size;
char f_name[];
{
    FILE *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(buff,t_size,y_size,f_pointer);
    fclose (f_pointer);
}

/*****/
hvdtoRGB()
/*****/
{
    double hd,f,vnrm,snrm,fm,fn,fk,r,g,b;

    hd = (double)ih/255.*6.;
    f = hd-(double)((int)hd);
    vnrm=(double)iv/255.;
    if (iv != 0) {
        snrm=(double)id/(double)iv;
    }
    else snrm = 0.0;
    fm = vnrm*(1.-snrm);
    fn = vnrm*(1.-(snrm*f));
    fk = vnrm*(1.-(snrm*(1.-f)));
    if((hd >= 0.)&&(hd < 1.)) {

```

```

        r=vnrm;
        g=fk;
        b=fm;
    }
    if((hd >= 1.)&&(hd < 2.)) {
        r=fn;
        g=vnrm;
        b=fm;
    }
    if((hd >= 2.)&&(hd < 3.)) {
        r=fm;
        g=vnrm;
        b=fk;
    }
    if((hd >= 3.)&&(hd < 4.)) {
        r=fm;
        g=fn;
        b=vnrm;
    }
    if((hd >= 4.)&&(hd < 5.)) {
        r=fk;
        g=fm;
        b=vnrm;
    }
    if(hd >= 5.) {
        r=vnrm;
        g=fm;
        b=fn;
    }

    ir = r * 255.;
    ig = g * 255.;
    ib = b * 255.;
}

```

```

/*****
program   linehvd.c
*****/

#include <stdio.h>
#include <math.h>
#define   tate  512
#define   yoko  512
#define   thp   128 /** BW hanten **/
unsigned char  r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];
unsigned char  h1[tate][yoko],v1[tate][yoko],d1[tate][yoko];
unsigned char  r2[tate][yoko],g2[tate][yoko],b2[tate][yoko];

main()
{
    char  fname1[64],fname2[64],fname3[64];
    char  fname1r[64],fname1h[64];
    char  fname1g[64],fname1v[64];
    char  fname1b[64],fname1d[64];
    int   i,j;
    int   lip;
    int   DCB[6];
    int   FMOD,SIZ,FNO,RD,RLMD,STN;

    printf("=== 512x512 image & scanline -> graphica ===\n");
    printf("ENTER! file name .h .v .d: ");
    scanf("%s",fname1);
    sprintf(fname1r,"%s.r",fname1);
    sprintf(fname1g,"%s.g",fname1);

```

# PIC 512-4.C

```

#include <stdio.h>
#include <math.h>

#define haba_x 1023
#define haba_y 1023
int   DCB[6],BUSY,LOCK,PRNT,ERR[3],ID,REM;
int   WIND[4],VIEW[4],READ,W[4],V[4],SIZ[2];
int   EXPOS;
int   STR,TFN;
int   i;
int   x,y;
main()
{
    for (i=0;i<6;i++) DCB[i]=0;
    opngd_(&DCB[0]);
    opnpct_(&DCB[0]);
    REM = 1; LOCK=0;
    printf("===== graphica 512x512 * 4 ---> PICTORO 2000 =====\n");
    remote_(&DCB[0],&REM);
    while(LOCK==0) {
        stspxt_(&DCB[0],&BUSY,&LOCK,&PRNT,&ERR[0],&ID);
        printf("BUSY=%d LOCK=%d PRNT=%d ERR:%d %d %d ID=%d\n",
            BUSY,LOCK,PRNT,ERR[0],ERR[1],ERR[2],ID);
    }
    READ = 1;
    wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
/**
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        WIND[0],WIND[1],WIND[2],WIND[3],
        VIEW[0],VIEW[1],VIEW[2],VIEW[3]);
**/

    /*** window-viewport settei *****/
    printf("ENTER ! ORG LL(x,y)");
    scanf("%d,%d",&x,&y);
    printf("\n");
    W[0]=x; W[1]=y; W[2]=x+haba_x; W[3]=y+haba_y;
    V[0]=0; V[1]=255; V[2]=2047; V[3]=2302;
/**
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        W[0],W[1],W[2],W[3],
        V[0],V[1],V[2],V[3]);
**/

    READ = 0;
    SIZ[0]=2;SIZ[1]=1;magsiz_(&DCB[0],&READ,&SIZ[0]);
    wndviw_(&DCB[0],&READ,&W[0],&V[0]);
    READ = 1;
    magsiz_(&DCB[0],&READ,&SIZ[0]);
    wndviw_(&DCB[0],&READ,&W[0],&V[0]);
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        W[0],W[1],W[2],W[3],
        V[0],V[1],V[2],V[3]);
    printf("SIZE:%d %d\n",SIZ[0],SIZ[1]);

    STR = 0; TFN = 1;
    frmtvo_(&DCB[0],&STR,&TFN);
    EXPOS = 0;
    print_(&DCB[0],&EXPOS);

    /*** moto no settei ni modosu *****/
    READ = 0;
    SIZ[0]=1;SIZ[1]=1;magsiz_(&DCB[0],&READ,&SIZ[0]);
    wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
    READ = 1;
    magsiz_(&DCB[0],&READ,&SIZ[0]);

```

```
wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
       WIND[0],WIND[1],WIND[2],WIND[3],
       VIEW[0],VIEW[1],VIEW[2],VIEW[3]);
printf("SIZE:%d %d\n",SIZ[0],SIZ[1]);

REM = 0;
remote_(&DCB[0],&REM);
stspxt_(&DCB[0],&BUSY,&LOCK,&PRNT,&ERR[0],&ID);
printf("BUSY=%d LOCK=%d PRNT=%d ERR:%d %d %d ID=%d\n",
       BUSY,LOCK,PRNT,ERR[0],ERR[1],ERR[2],ID);

c1sgd_(&DCB[0]);
```

```
}
```

```

/*****
  program   frmread.c
*****/

#include <stdio.h>
#include <math.h>
#define  tate  512
#define  yoko  512
unsigned char  r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];

main()
{
    char  fname1[64],fname2[64],fname3[64];
    char  fname1r[64],fname2r[64];
    char  fname1g[64],fname2g[64];
    char  fname1b[64],fname2b[64];
    int   i,j;

    int   DCB[6];
    int   FMOD,SIZ,FNO,RD,RLMD,STN;

    printf("=== 512x512 color image -> graphica ===\n");
    printf("ENTER! RGB file name .r .g .b: ");
    scanf("%s",fname1);
    printf("\n");
    printf("FRAME NUMBER? :");
    scanf("%d",&FNO);
    sprintf(fname1r,"%s.r",fname1);
    sprintf(fname1g,"%s.g",fname1);
    sprintf(fname1b,"%s.b",fname1);
    readingr(fname1r);
    readingg(fname1g);
    readingb(fname1b);

    opngd_(&DCB[0]);
    FMOD = SIZ = STN=1;
    STN = 0;
    frmtvo_(&DCB[0],&STN,&FNO);
    RD = 2;
    RLMD = 0;
    frmio_(&DCB[0],&FMOD,&SIZ,&FNO,&r1[0][0],&g1[0][0],&b1[0][0],&RD,&RLMD);
    STN = 1;
    frmtvo_(&DCB[0],&STN,&FNO);
    clsgd_(&DCB[0]);

}

/*****/
readingr(f_name)
/*****/
char      f_name[];
{
    FILE      *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(r1,tate,yoko,f_pointer);
    fclose (f_pointer);
}

/*****/
readingg(f_name)
/*****/
char      f_name[];
{

```

```

FILE          *f_pointer,*fopen();

if ((f_pointer = fopen(f_name,"r")) == NULL)
  { printf("\nfile read error : %s",f_name);
    exit(); }
printf("reading... %s\n",f_name);
fread(g1,tate,yoko,f_pointer);
fclose (f_pointer);
}
/*****/
readingb(f_name)
/*****/
char          f_name[];
{
FILE          *f_pointer,*fopen();

if ((f_pointer = fopen(f_name,"r")) == NULL)
  { printf("\nfile read error : %s",f_name);
    exit(); }
printf("reading... %s\n",f_name);
fread(b1,tate,yoko,f_pointer);
fclose (f_pointer);
}
/*****/
program      histimg.c

special (20,40) -- (490,335) no histogram
*****/

#include <stdio.h>
#include <math.h>
#define tate 512
#define yoko 512
/**
#define xorg 20
#define xwdt 470
#define yorg 40
#define ywdt 395
**/
#define xorg 200
#define xwdt 100
#define yorg 200
#define ywdt 100
#define thp 128 /** BW hanten **/
#define bbb 50
unsigned char r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];
unsigned char h1[tate][yoko],v1[tate][yoko],d1[tate][yoko];
long hr[256],hg[256],hb[256];
int hrr[256],hgg[256],hbb[256];

int ir,ig,ib,ih,iv,id;

main()
{
char fname1[64],fname2[64],fname3[64];
char fname1r[64],fname1h[64];
char fname1g[64],fname1v[64];
char fname1b[64],fname1d[64];
int i,j;
int lin;
int DCB[6];
int FMOD,SIZ,FNO,RD,RLMD,STN;

printf("=== 512x512 image & histgram -> graphica ===\n");
printf("ENTER! B/W file name : ");
scanf("%s",fname1);

```

```

/*****/
hvdrtorgb()
/*****/
{
    double  hd,f,vnrm,snrm,fn,fn,fk,r,g,b;

    hd = (double)ih/255.*6.;
    f = hd-(double)((int)hd);
    vnrm=(double)iv/255.;
    if (iv != 0) {
        snrm=(double)id/(double)iv;
    }
    else snrm = 0.0;
    fm = vnrm*(1.-snrm);
    fn = vnrm*(1.-(snrm*f));
    fk = vnrm*(1.-(snrm*(1.-f)));
    if((hd >= 0.)&&(hd < 1.)) {
        r=vnrm;
        g=fk;
        b=fm;
    }
    if((hd >= 1.)&&(hd < 2.)) {
        r=fn;
        g=vnrm;
        b=fm;
    }
    if((hd >= 2.)&&(hd < 3.)) {
        r=fm;
        g=vnrm;
        b=fk;
    }
    if((hd >= 3.)&&(hd < 4.)) {
        r=fm;
        g=fn;
        b=vnrm;
    }
    if((hd >= 4.)&&(hd < 5.)) {
        r=fk;
        g=fm;
        b=vnrm;
    }
    if(hd >= 5.) {
        r=vnrm;
        g=fm;
        b=fn;
    }

    ir = r * 255.;
    ig = g * 255.;
    ib = b * 255.;
}

```

```

/*****
program    koma30.c
*****/

```

```

#include <stdio.h>
#include <math.h>
#define  tate  512
#define  tate2 1024
#define  yoko  512
#define  yoko2 1280
unsigned char  r1[tate][yoko],g1[yoko][tate],b1[yoko][tate];

```



```

unsigned char  mr1[tate2][yoko2];
unsigned char  mg1[tate2][yoko2];
unsigned char  mb1[tate2][yoko2];
unsigned char  mr2[yoko2][tate2];
unsigned char  mg2[yoko2][tate2];
unsigned char  mb2[yoko2][tate2];
main()
{
    int  i,j,i1,i2,j1,j2,k;

    int  DCB[6];
    int  FMOD,SIZ,FNO,RD,RLMD,M,FUNCR,FUNCW,STR,x1,x2,y1,y2,PAGE;
    int  fnum,stp,xx1,xx2,yy1,yy2,xxd,yyd,x0,y0;
    int  RB,GB,BB;
    long NBYT;

    printf("==== 512x512x30 frm -->      ====\n");
    printf("====      image size 1280 x 1024      ====\n");
    printf("====      rotate 90 deg      ====\n");
    printf("ENTER start frame No.& step(1or2):");
    scanf("%d,%d",&fnum,&stp);
    printf("\n");

    opngd_(&DCB[0]);
    FMOD = SIZ = 1;
    RD = 2;
    RLMD = 0;
    RB=4; GB=2; BB=1;
    STR = 0;
    FNO = 1;

    frmtvo_(&DCB[0],&STR,&FNO);

    RD = 1;
    RLMD = 0;

    FNO = fnum;
    for (k=0; k<30; k++) {
        printf("FRAME = %d\n",FNO);

        STR=0; frmtvo_(&DCB[0],&STR,&FNO);
        frmio_(&DCB[0],&FMOD,&SIZ,&FNO,&r1[0][0],&g1[0][0],&b1[0][0],&RD,&RLMD);
        STR=1; frmtvo_(&DCB[0],&STR,&FNO);
        for (j=0;j<204;j++) {
            j1 = (int)(k/6)*204 + j;
            j2 = (int)((float)j*2.5);
            for (i=0;i<204;i++) {
                i1 = (k%6)*204 + i;
                i2 = (int)((float)i*2.5);
                /**
                mr1[j1][i1] = r1[j2][i2];
                mg1[j1][i1] = g1[j2][i2];
                mb1[j1][i1] = b1[j2][i2];
                **/
                /**
                mr1[j1][i1] = (unsigned char) (((float)r1[j2][i2]
                    +(float)r1[j2+1][i2] + (float)r1[j2][i2+1]
                    +(float)r1[j2+1][i2+1] ) / 4.);
                mg1[j1][i1] = (unsigned char) (((float)g1[j2][i2]
                    +(float)g1[j2+1][i2] + (float)g1[j2][i2+1]
                    +(float)g1[j2+1][i2+1] ) / 4.);
                mb1[j1][i1] = (unsigned char) (((float)b1[j2][i2]
                    +(float)b1[j2+1][i2] + (float)b1[j2][i2+1]
                    +(float)b1[j2+1][i2+1] ) / 4.);
                **/
                mr1[j1][i1] = (unsigned char) ((r1[j2][i2]

```

```

        +r1[j2+1][i2] + r1[j2][i2+1]
        +r1[j2+1][i2+1] ) / 4);
mg1[j1][i1] = (unsigned char) ((g1[j2][i2]
    +g1[j2+1][i2] + (float)g1[j2][i2+1]
    +g1[j2+1][i2+1] ) / 4);
mb1[j1][i1] = (unsigned char) ((b1[j2][i2]
    +b1[j2+1][i2] + b1[j2][i2+1]
    +b1[j2+1][i2+1] ) / 4);
    }
}
FNO = FNO + stp;
}
printf("rotate 90 deg -R-\n");
rotateR(yoko2,tate2);
printf("rotate 90 deg -G-\n");
rotateG(yoko2,tate2);
printf("rotate 90 deg -B-\n");
rotateB(yoko2,tate2);

FUNCW = 1280; FUNCRC = 1536;
NBYT = 1280*1024;
xx1=0;yy1=1279;xx2=1023;yy2=0;
STR=0; frmtvo_(&DCB[0],&STR,&FNO);

PAGE=0; pagsl_(&DCB[0],&PAGE);

dmast_(&DCB[0],&xx1,&yy1,&xx2,&yy2);
memsl_(&DCB[0],&RB);
imgio_(&DCB[0],&FUNCW,&NBYT,&mr2[0][0],&M);

dmast_(&DCB[0],&xx1,&yy1,&xx2,&yy2);
memsl_(&DCB[0],&GB);
imgio_(&DCB[0],&FUNCW,&NBYT,&mg2[0][0],&M);

dmast_(&DCB[0],&xx1,&yy1,&xx2,&yy2);
memsl_(&DCB[0],&BB);
imgio_(&DCB[0],&FUNCW,&NBYT,&mb2[0][0],&M);

FNO=1;
STR = 1;
frmtvo_(&DCB[0],&STR,&FNO);
c1sgd_(&DCB[0]);

}
/*****/
rotateR(xd,yd)
/*****/
int  xd,yd;
{
    int  i,j,i2,j2;

    for(j=0;j<yd;j++) {
        j2=yd-j-1;
        for (i=0;i<xd;i++) {
            i2=xd-i-1;
            mr2[i2][j] = mr1[j][i];
        }
    }
}
/*****/
rotateG(xd,yd)
/*****/
int  xd,yd;
{
    int  i,j,i2,j2;

```

```

    for(j=0;j<yd;j++) {
        j2=yd-j-1;
        for (i=0;i<xd;i++) {
            i2=xd-i-1;
            mg2[i2][j] = mg1[j][i];
        }
    }
}

```

```

/*****/
rotateB(xd,yd)
/*****/

```

```

int  xd,yd;
{
    int  i,j,i2,j2;

    for(j=0;j<yd;j++) {
        j2=yd-j-1;
        for (i=0;i<xd;i++) {
            i2=xd-i-1;
            mb2[i2][j] = mb1[j][i];
        }
    }
}

```

*pic2000.c*

```

#include <stdio.h>
#include <math.h>

```

```

#define haba_x 2047
#define haba_y 2559
int  DCB[6],BUSY,LOCK,PRNT,ERR[3],ID,REM;
int  WIND[4],VIEW[4],READ,W[4],V[4],SIZ[2];
int  EXPOS;
int  STR,TFN;
int  i;
int  x,y;

```

```

main()
{
    for (i=0;i<6;i++) DCB[i]=0;
    opngd_(&DCB[0]);
    opnpct_(&DCB[0]);
    REM = 1; LOCK=0;
    printf("==== graphica 2048x2560 ---> PICTORO 2000 =====\n");
    remote_(&DCB[0],&REM);
    while(LOCK==0) {
        stspxt_(&DCB[0],&BUSY,&LOCK,&PRNT,&ERR[0],&ID);
        printf("BUSY=%d LOCK=%d PRNT=%d ERR:%d %d %d ID=%d\n",
            BUSY,LOCK,PRNT,ERR[0],ERR[1],ERR[2],ID);
    }
    READ = 1;
    wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
/**
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        WIND[0],WIND[1],WIND[2],WIND[3],
        VIEW[0],VIEW[1],VIEW[2],VIEW[3]);
**/

    /*** window-viewport settei *****/
    printf("ENTER ! ORG LL(x,y)");
    scanf("%d,%d",&x,&y);
    printf("\n");
    W[0]=x; W[1]=y; W[2]=x+haba_x; W[3]=y+haba_y;
    V[0]=0; V[1]=0; V[2]=2047; V[3]=2559;
/**
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        W[0],W[1],W[2],W[3],
        V[0],V[1],V[2],V[3]);
**/

```

```

    READ = 0;
    SIZ[0]=1;SIZ[1]=1;magsiz_(&DCB[0],&READ,&SIZ[0]);
    wndviw_(&DCB[0],&READ,&W[0],&V[0]);
    READ = 1;
    magsiz_(&DCB[0],&READ,&SIZ[0]);
    wndviw_(&DCB[0],&READ,&W[0],&V[0]);
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
           W[0],W[1],W[2],W[3],
           V[0],V[1],V[2],V[3]);
    printf("SIZE:%d %d\n",SIZ[0],SIZ[1]);

    STR = 0; TFN = 1;
    frmtvo_(&DCB[0],&STR,&TFN);
    EXPOS = 0;
    print_(&DCB[0],&EXPOS);

    /*** moto no settei ni modosu *****/
    READ = 0;
    SIZ[0]=1;SIZ[1]=1;magsiz_(&DCB[0],&READ,&SIZ[0]);
    wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
    READ = 1;
    magsiz_(&DCB[0],&READ,&SIZ[0]);
    wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
           WIND[0],WIND[1],WIND[2],WIND[3],
           VIEW[0],VIEW[1],VIEW[2],VIEW[3]);
    printf("SIZE:%d %d\n",SIZ[0],SIZ[1]);

    REM = 0;
    remote_(&DCB[0],&REM);
    stspxt_(&DCB[0],&BUSY,&LOCK,&PRNT,&ERR[0],&ID);
    printf("BUSY=%d LOCK=%d PRNT=%d ERR:%d %d %d ID=%d\n",
           BUSY,LOCK,PRNT,ERR[0],ERR[1],ERR[2],ID);

    clsgd_(&DCB[0]);
}
/*****
    program    sfrmread.c
*****/

#include <stdio.h>
#include <math.h>
#define  Tate 512
#define  Yoko 512
unsigned char  r1[Tate][Yoko],g1[Tate][Yoko],b1[Tate][Yoko];

main()
{
    char  fname1[64],fname2[64],fname3[64];
    char  fname1r[64],fname2r[64];
    char  fname1g[64],fname2g[64];
    char  fname1b[64],fname2b[64];
    int   i,j;

    int   DCB[6];
    int   FMODE,SIZ,FNO,RD,RLMD,STN;
    int   is,ie;

    printf("==== n x 512x512 color image -> graphica ==== \n");
    printf("ENTER! RGB file name Onnn.r .g .b: ");
    scanf("%s",fname1);
    printf("\n");
    printf("FRAME NUMBER? (start,end):");
    scanf("%d,%d",&is,&ie);

```

```

RD = 2;
RLMD = 0;
opngd_(&DCB[0]);

for (i=is;i<=ie;i++) {
  if (1<i<10) {
    sprintf(fname1r,"%s000%d.r",fname1,i);
    sprintf(fname1g,"%s000%d.g",fname1,i);
    sprintf(fname1b,"%s000%d.b",fname1,i);
  }
  if ((10<=i)&&(i<100)) {
    sprintf(fname1r,"%s00%2d.r",fname1,i);
    sprintf(fname1g,"%s00%2d.g",fname1,i);
    sprintf(fname1b,"%s00%2d.b",fname1,i);
  }

  if ((100<=i)&&(i<1000)) {
    sprintf(fname1r,"%s0%3d.r",fname1,i);
    sprintf(fname1g,"%s0%3d.g",fname1,i);
    sprintf(fname1b,"%s0%3d.b",fname1,i);
  }
  readimgr(fname1r);
  readimgg(fname1g);
  readimgb(fname1b);
  FNO = i;
  FMOD = SIZ = STN=1;
  STN = 0;
  frmtvo_(&DCB[0],&STN,&FNO);

  frmio_(&DCB[0],&FMOD,&SIZ,&FNO,&r1[0][0],&g1[0][0],&b1[0][0],&RD,&RLMD);

  STN = 1;
  frmtvo_(&DCB[0],&STN,&FNO);
}

c1sgd_(&DCB[0]);

```

```

}

```

```

/*****/
readimgr(f_name)
/*****/
char      f_name[];
{
  FILE      *f_pointer,*fopen();

  if ((f_pointer = fopen(f_name,"r")) == NULL)
    { printf("\nfile read error : %s",f_name);
      exit(); }
  printf("reading... %s\n",f_name);
  fread(r1,tate,yoko,f_pointer);
  fclose (f_pointer);
}
/*****/
readimgg(f_name)
/*****/
char      f_name[];
{
  FILE      *f_pointer,*fopen();

  if ((f_pointer = fopen(f_name,"r")) == NULL)
    { printf("\nfile read error : %s",f_name);
      exit(); }
  printf("reading... %s\n",f_name);
  fread(g1,tate,yoko,f_pointer);
  fclose (f_pointer);
}

```

```
}
/*****/
readingb(f_name)
/*****/
char      f_name[];
{
    FILE      *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == .NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(b1,tate,yoko,f_pointer);
    fclose (f_pointer);
}
```

```

/*****
program   frmread-s.c
*****/

#include <stdio.h>
#include <math.h>
#define  tate  512
#define  yoko  512
unsigned char  r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];

main()
{
    char  fname1[64],fname2[64],fname3[64];
    char  fname1r[64],fname2r[64];
    char  fname1g[64],fname2g[64];
    char  fname1b[64],fname2b[64];
    int   i,j;

    int   DCB[6];
    int   FMOD,SIZ,FNO,FNOS,FNOE,RD,RLMD,STN;
    char  c1,c2,c3,c4;
    int   d1,d2,d3,d4;

    printf("=== 512x512 color image -> graphica  ===\n");
    printf("ENTER! RGB file name .r .g .b: ");
    scanf("%s",fname1);
    printf("\n");
    printf("FRAME NUMBER? (SATRT,END):");
    scanf("%d,%d",&FNO,&FNOE);
    printf("%d  %d \n",FNO,FNOE);
    for (i=FNO;i<=FNOE;i++) {
        d1 = i/1000;
        d2 = (i-1000*d1)/100;
        d3 = (i-1000*d1-100*d2)/10;
        d4 = i-1000*d1-100*d2-10*d3;
        c1 = d1 + 0x30;
        c2 = d2 + 0x30;
        c3 = d3 + 0x30;
        c4 = d4 + 0x30;
        printf("%c %c %c %c\n",c1,c2,c3,c4);
        sprintf(fname1r,"%s%c%c%c%c.r",fname1,c1,c2,c3,c4);
        sprintf(fname1g,"%s%c%c%c%c.g",fname1,c1,c2,c3,c4);
        sprintf(fname1b,"%s%c%c%c%c.b",fname1,c1,c2,c3,c4);
        readimgr(fname1r);
        readimgg(fname1g);
        readimgb(fname1b);

        FNO = i;
        opngd_(&DCB[0]);
        FMOD = SIZ = STN=1;
        STN = 0;
        frmtvo_(&DCB[0],&STN,&FNO);
        RD = 2;
        RLMD = 0;
        frmio_(&DCB[0],&FMOD,&SIZ,&FNO,&r1[0][0],&g1[0][0],&b1[0][0],&RD,&RLMD);
        STN = 1;
        frmtvo_(&DCB[0],&STN,&FNO);
        clsgd_(&DCB[0]);
    }
}

/*****/
readimgr(f_name)
/*****/
char      f_name[];

```

```

{
    FILE          *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(r1,tate,yoko,f_pointer);
    fclose (f_pointer);
}
/*****/
readingg(f_name)
/*****/
char          f_name[];
{
    FILE          *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(g1,tate,yoko,f_pointer);
    fclose (f_pointer);
}
/*****/
readingb(f_name)
/*****/
char          f_name[];
{
    FILE          *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(b1,tate,yoko,f_pointer);
    fclose (f_pointer);
}
/*****/
program      histhvd2.c
/*****/
#include <stdio.h>
#include <math.h>
#define      tate 512
#define      yoko 512
#define      thp  128 /** BW hanten **/
#define      bbb  50
unsigned char  r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];
unsigned char  h1[tate][yoko],v1[tate][yoko],d1[tate][yoko];
long          hr[256],hg[256],hb[256];
int           hrr[256],hgg[256],hbb[256];

int           ir,ig,ib,ih,iv,id;

main()
{
    char  fname1[64],fname2[64],fname3[64];
    char  fname1r[64],fname1h[64];
    char  fname1g[64],fname1v[64];
    char  fname1b[64],fname1d[64];
    int   i,j;
    int   lin;
    int   DCB[6];
    int   FMOD,SIZ,FNO,RD,RLMD,STN;
}

```



```

    b=vnrm;
}
if((hd >= 4.) && (hd < 5.)) {
    r=fk;
    g=fm;
    b=vnrm;
}
if(hd >= 5.) {
    r=vnrm;
    g=fm;
    b=fn;
}

ir = r * 255.;
ig = g * 255.;
ib = b * 255.;
}

```

```

/*****
program   koma20r.c
*****/

#include <stdio.h>
#include <math.h>
#define  tate  512
#define  tate2 1024
#define  yoko  512
#define  yoko2 1280
unsigned char  r1[tate][yoko],g1[yoko][tate],b1[yoko][tate];
unsigned char  mr1[tate2][yoko2];
unsigned char  mg1[tate2][yoko2];
unsigned char  mb1[tate2][yoko2];
unsigned char  mr2[yoko2][tate2];
unsigned char  mg2[yoko2][tate2];
unsigned char  mb2[yoko2][tate2];
main()
{
    int  i,j,i1,i2,j1,j2,k;

    int  DCB[6];
    int  FMOD,SIZ,FNO,RD,RLMD,M,FUNCR,FUNCW,STR,x1,x2,y1,y2;
    int  fnum,xx1,xx2,yy1,yy2,xxd,yyd,x0,y0;
    int  RB,GB,BB;
    long NBYT;

    printf("==== 512x512x20 frm  -->      ====\n");
    printf("====      image size 1280 x 1024  ====\n");
    printf("====      rotate 90 deg      ====\n");
    printf("ENTER start frame No.:");
    scanf("%d",&fnum);
    printf("\n");

    opngd_(&DCB[0]);
    FMOD = SIZ = 1;
    RD = 2;
    RLMD = 0;
    RB=4; GB=2; BB=1;
    STR = 0;
    FNO = 1;

    frmtvo_(&DCB[0],&STR,&FNO);

    RD = 1;
    RLMD = 0;

```

```

for (k=0; k<20; k++) {
    FNO = k + fnum;
    printf("FRAME = %d\n",FNO);

    frmio_(&DCB[0],&FMODE,&SIZ,&FNO,&r1[0][0],&g1[0][0],&b1[0][0],&RD,&RLMD);
    for (j=0;j<256;j++) {
        j1 = (int)(k/5)*256 + j;
        j2 = j*2;
        for (i=0;i<256;i++) {
            i1 = (k%5)*256 + i;
            i2 = i*2;
            mr1[j1][i1] = (unsigned char) ((r1[j2][i2]
                +r1[j2+1][i2] + r1[j2][i2+1]
                +r1[j2+1][i2+1] ) / 4);
            mg1[j1][i1] = (unsigned char) ((g1[j2][i2]
                +g1[j2+1][i2] + (float)g1[j2][i2+1]
                +g1[j2+1][i2+1] ) / 4);
            mb1[j1][i1] = (unsigned char) ((b1[j2][i2]
                +b1[j2+1][i2] + b1[j2][i2+1]
                +b1[j2+1][i2+1] ) / 4);
        }
    }

}

printf("rotate 90 deg -R-\n");
rotateR(yoko2,tate2);
printf("rotate 90 deg -G-\n");
rotateG(yoko2,tate2);
printf("rotate 90 deg -B-\n");
rotateB(yoko2,tate2);

FUNCW = 1280; FUNCN = 1536;
NBYT = 1280*1024;
xx1=0;yy1=1279;xx2=1023;yy2=0;
dmast_(&DCB[0],&xx1,&yy1,&xx2,&yy2);
memsl_(&DCB[0],&RB);
imgio_(&DCB[0],&FUNCW,&NBYT,&mr2[0][0],&M);

dmast_(&DCB[0],&xx1,&yy1,&xx2,&yy2);
memsl_(&DCB[0],&GB);
imgio_(&DCB[0],&FUNCW,&NBYT,&mg2[0][0],&M);

dmast_(&DCB[0],&xx1,&yy1,&xx2,&yy2);
memsl_(&DCB[0],&BB);
imgio_(&DCB[0],&FUNCW,&NBYT,&mb2[0][0],&M);

FNO=1;
STR = 1;
frmtvo_(&DCB[0],&STR,&FNO);
c1sgd_(&DCB[0]);

}
/*****/
rotateR(xd,yd)
/*****/
int  xd,yd;
{
    int  i,j,i2,j2;

    for(j=0;j<yd;j++) {
        j2=yd-j-1;
        for (i=0;i<xd;i++) {
            i2=xd-i-1;
            mr2[i2][j] = mr1[j][i];
        }
    }
}

```

```

    }
}
/*****/
rotateG(xd,yd)
/*****/
int  xd,yd;
{
    int  i,j,i2,j2;

    for(j=0;j<yd;j++) {
        j2=yd-j-1;
        for (i=0;i<xd;i++) {
            i2=xd-i-1;
            mg2[i2][j] = mg1[j][i];
        }
    }
}

```

```

/*****/
rotateB(xd,yd)
/*****/
int  xd,yd;
{
    int  i,j,i2,j2;

    for(j=0;j<yd;j++) {
        j2=yd-j-1;
        for (i=0;i<xd;i++) {
            i2=xd-i-1;
            mb2[i2][j] = mb1[j][i];
        }
    }
}

```

```

#include <stdio.h>
#include <math.h>

```

```

#define haba_x 1023
#define haba_y 1279
int  DCB[6],BUSY,LOCK,PRNT,ERR[3],ID,REM;
int  WIND[4],VIEW[4],READ,W[4],V[4],SIZ[2];
int  EXPOS;
int  STR,TFN;
int  i;
int  x,y;
main()
{
    for (i=0;i<6;i++) DCB[i]=0;
    opngd_(&DCB[0]);
    opnpct_(&DCB[0]);
    REM = 1; LOCK=0;
    printf("==== graphica 1024x1280 --> PICTORO 2000 =====\n");
    remote_(&DCB[0],&REM);
    while(LOCK==0) {
        stspxt_(&DCB[0],&BUSY,&LOCK,&PRNT,&ERR[0],&ID);
        printf("BUSY=%d LOCK=%d PRNT=%d ERR:%d %d %d ID=%d\n",
            BUSY,LOCK,PRNT,ERR[0],ERR[1],ERR[2],ID);
    }
    READ = 1;
    wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
/**
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        WIND[0],WIND[1],WIND[2],WIND[3],
        VIEW[0],VIEW[1],VIEW[2],VIEW[3]);
**/

/**** window-viewport settei ****/
    printf("ENTER ! ORG LL(x,y)");

```

pic1000.c

```

scanf("%d,%d",&x,&y);
printf("\n");
W[0]=x; W[1]=y; W[2]=x+haba_x; W[3]=y+haba_y;
V[0]=0; V[1]=0; V[2]=2047; V[3]=2559;
/**
printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
      W[0],W[1],W[2],W[3],
      V[0],V[1],V[2],V[3]);
**/

READ = 0;
SIZ[0]=2;SIZ[1]=1;magsiz_(&DCB[0],&READ,&SIZ[0]);
wndviw_(&DCB[0],&READ,&W[0],&V[0]);
READ = 1;
magsiz_(&DCB[0],&READ,&SIZ[0]);
wndviw_(&DCB[0],&READ,&W[0],&V[0]);
printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
      W[0],W[1],W[2],W[3],
      V[0],V[1],V[2],V[3]);
printf("SIZE:%d %d\n",SIZ[0],SIZ[1]);

STR = 0; TFN = 1;
frmtvo_(&DCB[0],&STR,&TFN);
EXPOS = 0;
print_(&DCB[0],&EXPOS);

/**** moto no settei ni modosu ****/
READ = 0;
SIZ[0]=1;SIZ[1]=1;magsiz_(&DCB[0],&READ,&SIZ[0]);
wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
READ = 1;
magsiz_(&DCB[0],&READ,&SIZ[0]);
wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
      WIND[0],WIND[1],WIND[2],WIND[3],
      VIEW[0],VIEW[1],VIEW[2],VIEW[3]);
printf("SIZE:%d %d\n",SIZ[0],SIZ[1]);

REM = 0;
remote_(&DCB[0],&REM);
stspxt_(&DCB[0],&BUSY,&LOCK,&PRNT,&ERR[0],&ID);
printf("BUSY=%d LOCK=%d PRNT=%d ERR:%d %d %d ID=%d\n",
      BUSY,LOCK,PRNT,ERR[0],ERR[1],ERR[2],ID);

clsgd_(&DCB[0]);
}

```

```

/*****
program    con4000c.c    full color
*****/

#include <stdio.h>
#include <math.h>
#define  tate  512
#define  yoko  512
unsigned char  r1[tate][yoko];
unsigned char  g1[tate][yoko];
unsigned char  b1[tate][yoko];
int  DCB[6];
int  FMOD,SIZ,FNO,RD,RLMD,M,FUNC,STR;
int  ISYN,MRLST,PAGE,PP;
int  RB,GB,BB;
int  conr,cong,conb;
long  NBYT;

main()
{
    int  i,j,k;
    int  pg,pgg;

    printf("==== GRAPHICA 4096x4096 nuri-tsubushi ===\n");
    printf("nuri-tsubushi value (r,g,b:0-255)?:");
    scanf("%d,%d,%d",&conr,&cong,&conb);
    printf("page number (0-7 ALL:9)?:");
    scanf("%d",&pgg);
    printf("\n");

    printf("GRAPHICA memory clear by %d %d %d\n",conr,cong,conb);
    for (i=0;i<yoko;i++) {
        for (j=0;j<tate;j++) {
            r1[j][i] = conr;
            g1[j][i] = cong;
            b1[j][i] = conb;
        }
    }
    opngd_(&DCB[0]);
    FMOD = SIZ = 1;
    RD = 2;
    RLMD = 0;
    RB=4; GB=2; BB=1;
    STR = 0;
    FUNC = 1280;
    FNO = 1;
    NBYT = tate * yoko;
    frmtvo_(&DCB[0],&STR,&FNO);
    ISYN = 0;
    syncm_(&DCB[0],&ISYN);
    if (pgg==9) {
        for (k=0;k<8;k++) {
            PAGE = k;
            memclear();
        }
    }
    else {
        PAGE = pgg;
        memclear();
    }
    FNO = 1;
    STR = 1;
    frmtvo_(&DCB[0],&STR,&FNO);
    clsgd_(&DCB[0]);
}

```

```

/*****/
memset()
/*****/
{
    int i,j,mm;
    int x1,x2,y1,y2,p1;

    pagsl_(&DCB[0],&PAGE);
    pagin_(&DCB[0],&PP);
    for (j=0;j<8;j++) {
        for (i=0;i<8;i++) {
            FNO = j*8 + i + 1;
            printf("PAGE: %d FRAME: %d\n",PP,FNO);
            x1 = i*yoko; y1 = (j+1)*tate-1;
            x2 = (i+1)*yoko-1; y2 = j*tate;
            dmast_(&DCB[0],&x1,&y1,&x2,&y2);
            memsl_(&DCB[0],&RB);
            imgio_(&DCB[0],&FUNC,&NBYT,&r1[0][0],&M);
            dmast_(&DCB[0],&x1,&y1,&x2,&y2);
            memsl_(&DCB[0],&GB);
            imgio_(&DCB[0],&FUNC,&NBYT,&g1[0][0],&M);
            dmast_(&DCB[0],&x1,&y1,&x2,&y2);
            memsl_(&DCB[0],&BB);
            imgio_(&DCB[0],&FUNC,&NBYT,&b1[0][0],&M);
        }
    }
}
/*****/
program   histrgb.c
/*****/

#include <stdio.h>
#include <math.h>
#define   tate  480
#define   yoko  512
#define   thp   128 /** BW hanten **/
unsigned char  r1[tate][yoko],g1[tate][yoko],b1[tate][yoko];
unsigned char  h1[tate][yoko],v1[tate][yoko],d1[tate][yoko];
long          hr[256],hg[256],hb[256];
int           hrr[256],hgg[256],hbb[256];

main()
{
    char  fname1[64],fname2[64],fname3[64];
    char  fname1r[64],fname1h[64];
    char  fname1g[64],fname1v[64];
    char  fname1b[64],fname1d[64];
    int   i,j;
    int   lin;
    int   DCB[6];
    int   FMOD,SIZ,FNO,RD,RLMD,STN;

    printf("=== 512x480 image & histogram -> graphica ===\n");
    printf("ENTER! file name .h .v .d: ");
    scanf("%s",fname1);
    printf("ENTER! FRAME NUMBER: ");
    scanf("%d",&FNO);
    printf("\n");
    sprintf(fname1r,"%s.r",fname1);
    sprintf(fname1g,"%s.g",fname1);
    sprintf(fname1b,"%s.b",fname1);
    sprintf(fname1h,"%s.h",fname1);
    sprintf(fname1v,"%s.v",fname1);
    sprintf(fname1d,"%s.d",fname1);
    readimg(r1,tate,yoko,fname1r);
    readimg(g1,tate,yoko,fname1g);

```

```

reading(b1,tate,yoko,fname1b);
reading(h1,tate,yoko,fname1h);
reading(v1,tate,yoko,fname1v);
reading(d1,tate,yoko,fname1d);
for (i=0;i<256;i++) {
    hr[i] = hg[i] = hb[i] = 0;
}

printf("histgram.....\n");
hist();

opngd_(&DCB[0]);
FMOD = SIZ = STN=1;
STN = 0;
frmtvo_(&DCB[0],&STN,&FNO);
RD = 2;
RLMD = 0;
frmio_(&DCB[0],&FMOD,&SIZ,&FNO,&r1[0][0],&g1[0][0],&b1[0][0],&RD,&RLMD);
STN = 1;
frmtvo_(&DCB[0],&STN,&FNO);
clsqd_(&DCB[0]);

```

```

}

```

```

/*****/

```

```

hist()

```

```

/*****/

```

```

{

```

```

    int i,j,k1,k2,lnum;
    int i1,i2,i3;
    int bbb=50;
    int on=255;
    int off=0;
    int hight=300;
    long mmm=0;

    for (j=0;j<tate;j=j++) {
        for (i=0;i<yoko;i=i++) {
            i1=h1[j][i]; hr[i1]=hr[i1]+1;
            i2=v1[j][i]; hg[i2]=hg[i2]+1;
            i3=d1[j][i]; hb[i3]=hb[i3]+1;
        }
    }

```

```

    for (i=0;i<256;i=i++) {
        if(mmm<hr[i]) mmm=hr[i];
        if(mmm<hg[i]) mmm=hg[i];
        if(mmm<hb[i]) mmm=hb[i];
    }

```

```

    for (i=0;i<256;i=i++) {
        hrr[i*2] = hrr[i*2+1] = hight*hr[i]/mmm;
        hgg[i*2] = hgg[i*2+1] = hight*hg[i]/mmm;
        hbb[i*2] = hbb[i*2+1] = hight*hb[i]/mmm;
    }

```

```

    for (i=0;i<yoko-1;i=i++) {
        k1 = tate - (int)hbb[i] - bbb;
        k2 = tate - (int)hbb[i+1] - bbb;
        plot2(i,k1,k2,on,off,on);
    }

```

```

    for (i=0;i<yoko-1;i=i++) {
        k1 = tate - (int)hgg[i] - bbb;
        k2 = tate - (int)hgg[i+1] - bbb;
        plot2(i,k1,k2,off,on,on);
    }

```

```

    }

    for (i=0;i<yoko-1;i=i++) {
        k1 = tate - (int)hrr[i] - bbb;
        k2 = tate - (int)hrr[i+1] - bbb;
        plot2(i,k1,k2,on,on,off);
        lnum = 512 - bbb;
        plot(i,lnum,on,on,on);
    }
}

/*****/
plot(x,y,r,g,b)
/*****/
int x,y,r,g,b;
{

    r1[y][x] = r;
    g1[y][x] = g;
    b1[y][x] = b;
}

/*****/
plot2(x0,y1,y2,r,g,b)
/*****/
int x0,y1,y2,r,g,b;
{
    int i,xx,yy,yh,yd;

    yd = abs(y1-y2);
    yh = yd/2;
    for (i=0;i<=yd;i++) {
        if (y1<y2) yy=y1+i;
        if (y1>=y2) yy=y1-i;
        if (i<=yd) xx=x0;
        if (i>yd) xx=x0+1;
        r1[yy][xx] = r;
        g1[yy][xx] = g;
        b1[yy][xx] = b;
    }
}

/*****/
reading(buff,t_size,y_size,f_name)
/*****/
unsigned char buff[];
int t_size,y_size;
char f_name[];
{
    FILE *f_pointer,*fopen();

    if ((f_pointer = fopen(f_name,"r")) == NULL)
        { printf("\nfile read error : %s",f_name);
          exit(); }
    printf("reading... %s\n",f_name);
    fread(buff,t_size,y_size,f_pointer);
    fclose (f_pointer);
}

/*****/
program koma20.c
*****/

#include <stdio.h>
#include <math.h>
#define tate 512

```



```

#define tate2 1024
#define yoko 512
#define yoko2 1280
unsigned char r1[tate2][yoko],g1[yoko][tate],b1[yoko][tate];
unsigned char mr1[tate2][yoko2];
unsigned char mg1[tate2][yoko2];
unsigned char mb1[tate2][yoko2];

main()
{
    int i,j,i1,i2,j1,j2,k;

    int DCB[6];
    int FMOD,SIZ,FNO,RD,RLMD,M,FUNCR,FUNCW,STR,x1,x2,y1,y2;
    int fnum,xx1,xx2,yy1,yy2,xxd,yyd,x0,y0;
    int RB,GB,BB;
    long NBYT;

    printf("==== 512x512x20 frm -->      ====\n");
    printf("==== image size 1280 x 1024 ====\n");
    printf("ENTER start frame No.:"");
    scanf("%d",&fnum);
    printf("\n");

    opngd_(&DCB[0]);
    FMOD = SIZ = 1;
    RD = 2;
    RLMD = 0;
    RB=4; GB=2; BB=1;
    STR = 0;
    FNO = 1;

    frmtvo_(&DCB[0],&STR,&FNO);

    RD = 1;
    RLMD = 0;

    for (k=0; k<20; k++) {
        FNO = k + fnum;
        printf("FRAME = %d\n",FNO);

        frmio_(&DCB[0],&FMOD,&SIZ,&FNO,&r1[0][0],&g1[0][0],&b1[0][0],&RD,&RLMD);
        for (j=0;j<256;j++) {
            j1 = (int)(k/5)*256 + j;
            j2 = j*2;
            for (i=0;i<256;i++) {
                i1 = (k%5)*256 + i;
                i2 = i*2;
                mr1[j1][i1] = r1[j2][i2];
                mg1[j1][i1] = g1[j2][i2];
                mb1[j1][i1] = b1[j2][i2];
            }
        }
    }

    FUNCW = 1280; FUNCR = 1536;
    NBYT = 1280*1024;
    xx1=0;yy1=1023;xx2=1279;yy2=0;
    dmast_(&DCB[0],&xx1,&yy1,&xx2,&yy2);
    memsl_(&DCB[0],&RB);
    imgio_(&DCB[0],&FUNCW,&NBYT,&mr1[0][0],&M);

    dmast_(&DCB[0],&xx1,&yy1,&xx2,&yy2);
    memsl_(&DCB[0],&GB);
    imgio_(&DCB[0],&FUNCW,&NBYT,&mg1[0][0],&M);
}

```

```
dmast_(&DCB[0],&xx1,&yy1,&xx2,&yy2);
mems1_(&DCB[0],&BB);
imgio_(&DCB[0],&FUNCW,&NBYT,&mb1[0][0],&M);
```

```
FNO=1;
STR = 1;
frmtvo_(&DCB[0],&STR,&FNO);
clsgd_(&DCB[0]);
```

```
}
#include <stdio.h>
#include <math.h>
```

```
#define haba_x 1023
#define haba_y 1279
int DCB[6],BUSY, LOCK, PRNT,ERR[3],ID,REM;
int WIND[4],VIEW[4],READ,W[4],V[4];
int EXPOS;
int STR,TFN;
int i;
int x,y;
main()
{
    for (i=0;i<6;i++) DCB[i]=0;
    opngd_(&DCB[0]);
    opnpct_(&DCB[0]);
    REM = 1; LOCK=0;
    printf("===== graphica 1024x1280 ---> PICTORO =====\n");
    remote_(&DCB[0],&REM);
    while(LOCK==0) {
        stspxt_(&DCB[0],&BUSY,&LOCK,&PRNT,&ERR[0],&ID);
        printf("BUSY=%d LOCK=%d PRNT=%d ERR:%d %d %d ID=%d\n",
            BUSY,LOCK,PRNT,ERR[0],ERR[1],ERR[2],ID);
    }
    READ = 1;
    wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
    /**/
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        WIND[0],WIND[1],WIND[2],WIND[3],
        VIEW[0],VIEW[1],VIEW[2],VIEW[3]);
    *****/

    /**/ window-viewport settei ****/
    printf("ENTER ! ORG LL(x,y)");
    scanf("%d,%d",&x,&y);
    printf("\n");
    W[0]=x; W[1]=y; W[2]=x+haba_x; W[3]=y+haba_y;
    V[0]=0; V[1]=0; V[2]=1023; V[3]=1279;
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        W[0],W[1],W[2],W[3],
        V[0],V[1],V[2],V[3]);

    READ = 0;
    wndviw_(&DCB[0],&READ,&W[0],&V[0]);
    READ = 1;
    wndviw_(&DCB[0],&READ,&W[0],&V[0]);
    printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        W[0],W[1],W[2],W[3],
        V[0],V[1],V[2],V[3]);

    STR = 0; TFN = 1;
    frmtvo_(&DCB[0],&STR,&TFN);
    EXPOS = 0;
    print_(&DCB[0],&EXPOS);
```

pic1000.c

```

/**** moto no settei ni modosu ****/
  READ = 0;
  wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
  READ = 1;
  wndviw_(&DCB[0],&READ,&WIND[0],&VIEW[0]);
  printf("WIND:%d %d %d %d VIEW:%d %d %d %d\n",
        WIND[0],WIND[1],WIND[2],WIND[3],
        VIEW[0],VIEW[1],VIEW[2],VIEW[3]);

  REM = 0;
  remote_(&DCB[0],&REM);
  stspxt_(&DCB[0],&BUSY,&LOCK,&PRNT,&ERR[0],&ID);
  printf("BUSY=%d LOCK=%d PRNT=%d ERR:%d %d %d ID=%d\n",
        BUSY,LOCK,PRNT,ERR[0],ERR[1],ERR[2],ID);

  clsgd_(&DCB[0]);
}
/*****/
reading(buff,t_size,y_size,f_name)
/*****/
unsigned char buff[];
int          t_size,y_size;
char         f_name[];
{
  FILE          *f_pointer,*fopen();

  if ((f_pointer = fopen(f_name,"r")) == NULL)
    { printf("\nfile read error : %s",f_name);
      exit(); }
  printf("reading... %s\n",f_name);
  fread(buff,t_size,y_size,f_pointer);
  fclose (f_pointer);
}
/*****/
writeimg(buff,t_size,y_size,f_name)
/*****/
unsigned char buff[];
int          t_size,y_size;
char         f_name[];
{
  FILE          *f_pointer,*fopen();

  if ((f_pointer = fopen(f_name,"w")) == NULL)
    { printf("\nfile create error : %s",f_name);
      exit(); }
  printf("writing... %s\n",f_name);
  fwrite(buff,t_size,y_size,f_pointer);
  fclose (f_pointer);
}

```

```

/*****
program    con4000.c
*****/

#include <stdio.h>
#include <math.h>
#define  tate  512
#define  yoko  512
unsigned char  r1[tate][yoko];
int  DCB[6];
int  FMODE,SIZ,FNO,RD,RLMD,M,FUNC,STR;
int  ISYN,MRLST,PAGE,PP;
int  RB,GB,BB;
int  con;
long  NBYT;

main()
{
    int  i,j,k;
    int  pg,pgg;

    printf("==== GRAPHICA 4096x4096 nuri-tsubushi ===\n");
    printf("nuri-tsubushi value (0-255)?:" );
    scanf("%d",&con);
    printf("page number (0-7 ALL:9)?:" );
    scanf("%d",&pgg);
    printf("\n");

    printf("GRAPHICA memory clear by %d\n",con);
    for (i=0;i<yoko;i++) {
        for (j=0;j<tate;j++) {
            r1[j][i] = con;
        }
    }
    opngd_(&DCB[0]);
    FMODE = SIZ = 1;
    RD = 2;
    RLMD = 0;
    RB=4; GB=2; BB=1;
    STR = 0;
    FUNC = 1280;
    FNO = 1;
    NBYT = tate * yoko;
    frmtvo_(&DCB[0],&STR,&FNO);
    ISYN = 0;
    syncm_(&DCB[0],&ISYN);
    if (pgg==9) {
        for (k=0;k<8;k++) {
            PAGE = k;
            memclear();
        }
    }
    else {
        PAGE = pgg;
        memclear();
    }
    FNO = 1;
    STR = 1;
    frmtvo_(&DCB[0],&STR,&FNO);
    clsgd_(&DCB[0]);
}

/*****/
memclear()
/*****/
{

```

```

int i,j,mm;
int x1,x2,y1,y2,p1;

pagsl_(&DCB[0],&PAGE);
pagin_(&DCB[0],&PP);
for (j=0;j<8;j++) {
  for (i=0;i<8;i++) {
    FNO = j*8 + i + 1;
    printf("PAGE: %d  FRAME: %d\n",PP,FNO);
    x1 = i*yoko; y1 = (j+1)*tate-1;
    x2 = (i+1)*yoko-1; y2 = j*tate;
    dmast_(&DCB[0],&x1,&y1,&x2,&y2);
    memsl_(&DCB[0],&RB);
    imgio_(&DCB[0],&FUNC,&NBYT,&r1[0][0],&M);
    dmast_(&DCB[0],&x1,&y1,&x2,&y2);
    memsl_(&DCB[0],&GB);
    imgio_(&DCB[0],&FUNC,&NBYT,&r1[0][0],&M);
    dmast_(&DCB[0],&x1,&y1,&x2,&y2);
    memsl_(&DCB[0],&BB);
    imgio_(&DCB[0],&FUNC,&NBYT,&r1[0][0],&M);
  }
}
}
/** HISTGRAM          *****/
/***** hist.c *****/

```

```

#include <stdio.h>
#define tate 512
#define yoko 512

```

```

unsigned char  img1[tate][yoko],im;
long          h[256];
char          l[64];

```

```

main()
{

```

```

  char        fname1[64];
  FILE        *fp1,*fopen();
  int         i,j;
  long        tt;

```

```

  printf("=== HISTGRAM ===\n");
  printf("\nENTER! <512x512 file name >:");
  scanf ("%s",fname1);

```

```

  printf("reading... %s\n",fname1);
  fp1 = fopen(fname1,"r");
  fread(img1,yoko,tate,fp1);
  fclose(fp1);

```

```

  for (j=0;j<256;j++) h[j] = 0;

```

```

  printf("processing... \n");
  for (j=0;j<tate;j++) {
    for (i=0;i<yoko;i++) {
      im = img1[j][i];
      h[im] = h[im] + 1;
    }
  }

```

```

  tt = 0;
  printf("HISTGRAM OF %s\n",fname1);
  for (j=0;j<256;j++) {
    tt = tt + h[j]*j;
    bar(j);
    printf("%3d -> %6d :%s\n",j,h[j],l);
  }
}

```

```

        printf("\n>> AVERAGE: %3d\n",tt/tate/yoko);
    }
    /*****/
    bar(j)
    /*****/
    int j;
    {
        int i;
        long k;

        k = 512*512/64/8;
        for (i=0;i<64;i++) {
            l[i] = '.';
            if ( (k*(i+1)) <= h[j] ) l[i] = '#';
        }
    }
}
/*****
program    knj.c
*****/

#include <stdio.h>
#include <math.h>
#define tate 1024
#define yoko 1280
unsigned char m1[tate][yoko],m2[yoko][tate];

main()
{
    int i,j,k;

    int DCB[6];
    int FNO,SIZ,FNO, RD, RLMD,M, FUNCN, FUNCW, STR,x1,x2,y1,y2;
    int BUFF[64],IBUF[64];
    int LX[8],LY[8];
    short KSTR[2];
    char JSTA[4];
    float XX[1],YY[1];
    int PAGE, MEMNO, KIND, ISW, NN, IDIR, IR, IG, IB;
    int xx1,xx2,yy1,yy2,xxd,yyd,x0,y0;
    int RB,GB,BB;
    long NBYT;

    printf("==== KANJI HYOUJI TEST ==== \n");
    opngd_(&DCB[0]);
    FNO = 1;

    STR = 0;
    frmtvo_(&DCB[0],&STR,&FNO);

    PAGE=0;
    pagsl_(&DCB[0],&PAGE);

    MEMNO=7;
    memsl_(&DCB[0],&MEMNO);

    IR=255,IG=255,IB=0;
    vcolr_(&DCB[0],&IR,&IG,&IB);

    /****
    JSTA[0]=0x31;JSTA[1]=0x31;NN=2;
    vctae_(&DCB[0],&JSTA[0],&NN,&IBUF[0]);
    ****/

    k=0x3021;
    while(k<0x737f) {
        STR = 0;
        frmtvo_(&DCB[0],&STR,&FNO);

```

```
LX[0]=LX[3]=LX[4]=0;LX[1]=LX[2]=511;
LY[0]=LY[1]=LY[4]=250;LY[2]=LY[3]=280;
NN=5;
epoly_(&DCB[0],&LX[0],&LY[0],&NN,&BUFF[0]);
```

```
KIND=7;
vkind_(&DCB[0],&KIND);
XX[0]=0.;YY[0]=250.;NN=1;
dlina_(&DCB[0],&XX[0],&YY[0],&NN,&BUFF[0]);
KIND=0;
vkind_(&DCB[0],&KIND);
for (i=0;i<17;i++) {
    KSTR[0]=k;
    IDIR=0;NN=1;
    kanji_(&DCB[0],&KSTR[0],&IDIR,&NN,&IBUF[0]);
    k++;
}
STR=1;
frmtvo_(&DCB[0],&STR,&FNO);
sleep(1);
}
c!sgd_(&DCB[0]);
```

```
}
/*****
program memint.c
*****/
```

```
#include <stdio.h>
#include <math.h>
#define tate 512
#define yoko 512
unsigned char r1[tate][yoko];
int DCB[6];
int FMOD,SIZ,FNO,RD,RLMD,M,FUNC,STR;
int ISYN,MRLST,PAGE,PP;
int RB,GB,BB;
int con;
long NBYT;
```

```
main()
{
    int i,j,k;
    int pg,pgg;

    printf("==== GRAPHICA memory init RGB+C ===\n");
    opngd_(&DCB[0]);
    FMOD = SIZ = 1;
    STR = 0;
    FUNC = 1280;
    FNO = 1;
    frmtvo_(&DCB[0],&STR,&FNO);
    ISYN = 0;
    syncm_(&DCB[0],&ISYN);
    for (k=0;k<8;k++) {
        PAGE = k;
        pagsl_(&DCB[0],&PAGE);
        pagin_(&DCB[0],&PP);
        printf("PAGE: %d FRAME: %d\n",PP,FNO);
        initl_(&DCB[0]);
    }
    STR = 1; FNO = 1;
    frmtvo_(&DCB[0],&STR,&FNO);
    c!sgd_(&DCB[0]);
```

```
}
/*****
```