

KDrasterコンバート

(ベクター/ラスター変換)

H2xRS

H2xRSC



Kernel Computer System
カーネルコンピュータシステム株式会社

本社：パッケージ販売部
〒221-0056
横浜市神奈川区金港町 6-3 横浜金港町ビル
TEL：045-442-0500 FAX：045-442-0501
URL：<http://www.kernelcomputer.co.jp>

特 長

- HP-GL/2 ファイルからラスターデータへの変換を行いません。
- コマンドラインから入出力ファイル名をキーインする事で実行されます。(標準入出力が利用可能)
- 実行時各入出力フォーマットの属性パラメータやオプションパラメータを設定できます。
- スケール、回転、原点移動、センタリングなどの処理が可能。
- Windows の True Type Font 名を指定して、文字パターン生成が可能。使用可能な True Type Font 名は D FPOP 体、D F PPOP 体、Arial、Times New Roman、MS Pゴシック、MS P明朝、MS ゴシック、MS 明朝、MS UI Gothic となります。
- フォルダ毎の一括変換が可能
- 文字コードは、JIS ASCII、半角カタカナ、漢字のみの対応となります。

動作環境

| 機 種 | OS |
|------------|--|
| SUN/SPARC | Solaris2.5.1以降 |
| HP9000 | HP-UX11i V2以降 |
| IBM/RS6000 | AIX4.1.5以降 |
| Windows | 2000 / XP / Server 2003 / Vista / Server 2008 / 7 / Server 2008 R2 |
| LINUX | x86系32ビット版 |

製品一覧

| 製品名 | 製品仕様 | 標準価格 |
|-------------|---|------|
| H 2 2 R S | A0対応版(A0/600DPI) (JIS第1水準漢字対応) | 35万円 |
| H 2 3 R S | 長尺対応版(6A0/3000DPI) ※1 (JIS第1水準漢字対応) | 45万円 |
| H 2 3 R S C | 長尺対応版(6A0/3000DPI) ※1 モノクロ・カラー併用版 (JIS第1水準漢字対応) | 60万円 |

※1:10mまで出力可能です。

オプションソフト

| 製品名 | 製品概要 | 標準価格 |
|-----------------------------------|---|------|
| KDCONV_DIR (階層構造一括変換) | 階層構造の各フォルダ内に散らばっている必要なファイルを拡張子で指定し、下層フォルダまで一括変換します。 | 15万円 |
| KDauto_trans (標準版) (自動監視・自動出力) | 監視フォルダへ入力データを格納する事で自動的にデータ変換を行います。変換結果は印刷、ファイル転送、指定した別のフォルダへのファイル出力ができます。 | 20万円 |
| JIS 第2水準ベクターフォント | JIS 第2水準フォントファイルです。 文字を線分化する際に使用します。 | 30万円 |

操作方法

H22RS, H23RS, H23RSC は、コマンドラインアプリケーションです。
通常のコマンドと同じようにコマンドラインから実行したり、バッチファイルやシェルスクリプトに組み込む事ができます。
Windows の場合は、「コマンドプロンプト」で DOS ウィンドウを開いて、その中で実行します。
変換時にオプションを指定したり、属性データ変換パラメータファイルの内容を変更する事で様々な変換処理が可能となります。

HP-GL/2 からラスターデータに変換するには、次のような形式でコマンドラインに入力します。

```
実行モジュール 入力ファイル -0 出力ファイル -各オプション
```

[操作例]

H2xRS を使用して HP-GL/2 ファイル「sample.h2」を TIFF ファイル「sample.tif」に変換する場合は以下のように入力して変換します。

```
A> H2xRS sample.h2 -0sample.tif -Zrasl_out.atr  
ただいま、ファイル変換中です。
```

```
H2xRS 変換終了
```

H2xRS : 実行モジュール名 (例:H22RS)
sample.h2 : 入力ファイル名
sample.tif : 出力ファイル名
rasl_out.atr : 出力属性ファイル名

-0 : 変換結果を出力するファイル名を指定するオプション。
マイナス記号とアルファベットの「オー」です。
-0 と変換後のファイル名の間には空白を入れないようにします。
変換前のファイル名と変換後のファイル名に同じものを指定する事はできません。

- ※ オプション指定は大文字でも小文字でもかまいませんが、全角文字は使えません。
- ※ オプションの詳細についてはオプションの項目を参照下さい。
- ※ H2xRSC も同様。

オプション

- A : 指定した用紙サイズ(-P)に収まるように、縦横比を維持してスケーリングします。
- AJ : 指定した用紙サイズ(-P)を越える場合のみ、その用紙サイズに収まるように、図面を縮小して変換します。
- AL : 入力データサイズから用紙サイズ(A系列)に合わせてセンタリングを行います。
- M : 指定された出力ファイルの最後へ変換結果を追加(マルチページ)して出力を行います。出力時のラスターファイルがマルチページに対応したフォーマット(TIFF, FX-RASTER等)
- MFILE : 入力がマルチページファイルの場合、ページ単位に分割して出力します。各出力ファイル名は、指定した出力ファイル名に番号を付加したものになります。
<出力例>
H22RS sample.plt -Osample.tif -MFILE と入力した場合
sample_1.tif, smaple_2.tif, sample_3.tif…… と出力されます。
- P[paper] : 用紙サイズを番号で指定します。(0:A0 1:A1 2:A2 3:A3 4:A4 など)
- LS : 用紙サイズがA0版を超える場合に指定します。実際の指示は-Pオプションによって行います。(H22RSのみ)
- LD : 出力解像度が600DPIを超える場合に指定します。実際の指示は出力属性パラメータ(OUT_RESOLUTION)で行います。(H22RSのみ)
- MG[type] : ベクターデータ(HP-GL/2)とラスターデータ(HP RTL)の合成方法を指定します。ベクターデータ同士、ラスターデータ同士の合成には影響しません。(H22RS, H23RSのみ)
- TR[type] : カラー変換を行なう際の処理の方法を設定します。(H23RSCのみ)
- TRMODE : カラー処理を-TRの設定内容で変換するようにします。(H23RSCのみ)
- CMAP : フルカラーをインデックスカラーに変換します。(-TRパラメータ設定時に有効)
- MONO : カラー出力でなく、モノクロ(2値)に変換します。(H23RSCのみ)
- R[rotation] : 出力時90, 180, 270度回転します。(反時計方向)
- S[scale] : スケール値を指定します。
- X[xoff] : 水平方向のオフセット量を、mm単位の値で指定します。
- Y[yoff] : 垂直方向のオフセット量を、mm単位の値で指定します。
- BATCH[batch_file] : バッチファイルを使って、複数ファイルを一括処理します。

属性デ - タ変換パラメ - タファイル

属性データ変換パラメータファイルには、初期状態では

HP-GL/2 入力データ用 hp2_in. atr
ラスタ (モノクロ) 出力データ用 rasl_out. atr
ラスタ (カラー) 出力データ用 rascl_out. atr (H23RSC のみ)

があり、この内容を変更する事で種々な変換が可能となります。

■入力属性データ変換パラメータ

IN_VERSION : 入力データフォーマットを変更
GENTEN_MODE : 入力データの原点調整
VECTOR_ROT : Vector/Raster 混在時の回転を調整
CLIPAREA_MODE : クリップ命令の対応を調整
IN_SYMWD, INSYMHT : 入力データの文字高さ、文字幅を倍尺で指定
PENCOLOR : 入力データのペン番号と色番号を変更
DEF_LINE_CAP : 先端形状を設定
WINDOWS_FONT : 文字を WINDOWS(TrueType) フォント (MS 明朝や MS ゴシックなど) に
応じた輪郭と塗りで出力
WINDOWS_FONT_HT : WINDOWS フォントの高さを入力データの文字高に対して調整 ※
WINDOWS_FONT_WD : WINDOWS フォントの幅を入力データの文字幅に対して調整 ※
WINDOWS_FONT_SP : 文字と文字の間隔を任意に指定 ※
※WINDOWS_FONT パラメータ入力時のみ有効

HP-GL/2 入力データ用データ変換属性パラメータファイルのサンプル (hp2_in. atr)

```
#####hp2_in. atr#####  
#HP plotter model no. (ANY, HP-GL/2, HP7580X (HP7580A, HP7585A))  
# HP7590X (HP7595X, HP7596X)  
IN_VERSION=HP-GL/2  
#IN_VERSION=HP7580X  
#IN_VERSION=HP7590X  
#pen & line_color table  
PENCOLOR=(P1=1, P2=2, P3=3, P4=4, P5=5, P6=6, P7=7, P8=8)  
#symbol_height_scale  
IN_SYMHT = 1.0  
#symbol_width_scale  
IN_SYMWD = 1.0  
#pen & layer table (DXF_OUT, IGES_OUT)  
#PENLAYER=(P1=L1, P2=L2, P3=L3, P4=L4, P5=L5, P6=L6, P7=L7, P8=L8)  
#SP paper feed mode (Y:paper feed N:not pen)  
#SP_PAPERFEED = N  
# coordinates origin point (0:Lower_Left(default), 1:Hpmode_no ,  
2:Center)  
#GENTEN_MODE = 1  
#fill mode (0:output fill, -1:input fill(default), -2:not fill)  
#FILL_MODE = 0  
# original position (V:HP-GL/2(default), R:HP-RTL)  
#ORIGIN_POSITION = V  
#ORIGIN_POSITION = R  
# convert data auto rotation (H:horizontal(default), V:vertical)  
#A4_VH = V  
#A3_VH = V  
#A2_VH = V  
#A1_VH = V  
#A0_VH = V  
# vector data rotation (0:OFF, 1:ON(same as raster direction))  
#VECTOR_ROT = 0  
# RGB color/pen cross conversion table (max 256 tables)  
#RGB_PEN=(P1=0, 0, 0, P2=0, 1000, 1000, P3=1000, 0, 1000, P4=0, 0, 1000)  
#RGB_PEN=(P5=1000, 1000, 0, P6=0, 1000, 0, P7=1000, 0, 0, P8=1000, 1000, 1000)  
# Pen width / pen cross conversion table (max 256 tables)  
#WIDTH_PEN=(P1=0. 1, P2=0. 2, P3=0. 3, P4=0. 4, P5=0. 5, P6=0. 6, P7=0. 7, P8=0. 8)  
#clip command on/off mode (3: IP&PS(default), 0:clip off, 1:only IP, 2:only PS)  
# (4:mode=0 & last IW, 5:IW ignore)  
#CLIPAREA_MODE=0  
#clip area expand (UNIT:mm, default:0)
```

```
#CLIP_EXPAND_LEFT = -2.0
#CLIP_EXPAND_BOTTOM = -2.0
#CLIP_EXPAND_RIGHT = 2.0
#CLIP_EXPAND_TOP = 2.0
# HP-GL/2 mode, Line CAP (0:cut, 1:square, 2:circle)
DEF_LINE_CAP = 2
# true Type Font (Font_name, Height_scale, Width_scale, moji_space)
#WINDOWS_FONT=MS 明朝
#WINDOWS_FONT_HT=1.0
#WINDOWS_FONT_WD=1.3
#WINDOWS_FONT_SP=0.5
##### Raster conversion #####
# quality type (-1,0(default),1,2)
QUALITY_TYPE = 0
# sikkii_value (0 --> 255) (only use QUARITY_TYPE=1 (default:128))
SIKII_VALUE = 128
#
##### end hp2_in.atr#####
```

■出力属性データ変換パラメータ

| | |
|------------------|---|
| OUT_VERSION | : 出力データフォーマットを指定 |
| ATR_LINEWIDTH | : 線幅の有効区分を指定。(Y:属性ファイル/N:HP-GL/2 データ) |
| COLORWIDTH | : 出力データの線幅を指定 |
| OUT_RESOLUTION | : 出力データの解像度を指定 |
| OUTPUT_SIZE_X(Y) | : 出力データの出力範囲を指定 |
| CUT_SPAC_AREA | : 自動スケール時、片軸の余白カットの設定 |
| COLORRGB | : カラー出力 (OUT_VERSION=FX_RASTER のみ可能) の設定 (H23RS の場合) 入力データにカラー指定がない HP-GL, HP-GL/2 をカラー出力 (H23RSC の場合) |
| SYMBOL_LINEWIDTH | : 入力データの中の文字描画の線幅の変更 |
| A_PAPER_OUT | : 出力データを A 系列の用紙サイズに設定 |
| OUTPUT_DOT_X(Y) | : 出力サイズをドット指定。(H22RS のみ) |
| PAPER_SCALE | : 長尺用紙のスケールと回転の自動処理の ON・OFF 設定 ※1 |
| XX_OUT_SIZE_X(Y) | : 自動処理のスケールの設定 ※1 ※2 |
| XX_OUT_ROTATE | : 自動処理の回転の設定 ※1 ※2 |

※1 H23RS, H23RSC のみ設定可能。

※2 XX は用紙サイズ。(例 1:A0_OUT_SIZE_X, 例 2:A1_OUT_ROTATE)

- ・出力属性データ変換属性パラメータファイルのサンプル(モノクロ/長尺用) (rasl_out.atr)

```
#####rasl_out.atr#####
# format type (TIFF, SUN_RASTER, XWD, MMR, MR, MH, CALSG4, BMP etc.)
#OUT_VERSION=TIFF_NON
#OUT_VERSION=TIFF_PACK
#OUT_VERSION=TIFF_G31D
#OUT_VERSION=TIFF_G3_MH
#OUT_VERSION=TIFF_G3_MR
OUT_VERSION=TIFF_G4
#OUT_VERSION=SUN_RASTER
#OUT_VERSION=SUN_MMR
#OUT_VERSION=XWD
#OUT_VERSION=MMR
#OUT_VERSION=MR
#OUT_VERSION=MH
#OUT_VERSION=CALSG4
#OUT_VERSION=IOCA_1BMMMR
#OUT_VERSION=IOCA_NON
#OUT_VERSION=IOCA_G4MMR
#OUT_VERSION=BMP
#OUT_VERSION=MIEL
#OUT_VERSION=FX_RASTER
#OUT_VERSION=HPRTL_ROW
#OUT_VERSION=HPRTL_NON
#OUT_VERSION=HPRTL_PACK
#OUT_VERSION=HPRTL_MH
#OUT_VERSION=HPRTL_MR
#OUT_VERSION=HPRTL_MMR
#symbol height_scale (0.01 --> 99.99)
OUT_SYMHT=1.0
# symbol width_scale (0.01 --> 99.99)
OUT_SYMWD=1.0
# symbol space (unit:mm) (0.0 --> 99.99)
OUT_SYMSP=0.0
#line type pattern set(pattern_length * scale_factor) (unit:mm)
#DOT_LINE = 1.0, 1.0
#DASH_LINE = 2.0, 1.0
#CENTER_LINE = 10.0, 1.0, 1.0
#DIVIDE_LINE = 10.0, 1.0, 1.0
# linewidth available (Y:attribute(default), N:input data)
ATR_LINEWIDTH = N
# color/width cross conversion table (max 256 tables)
#COLORWIDTH=(C0=0.1, C1=0.1, C2=0.2, C3=0.3, C4=0.4, C5=0.5, C6=0.3, C7=0.7, C8=0.8)
#COLORWIDTH=(C9=0.1, C10=0.1, C11=0.1, C12=0.2, C13=0.3, C14=0.4, C15=0.5, C16=0.6)
#COLORWIDTH=(C128=0.01, C254=0.01, C255=0.01)
# all vector (without symbol) linewidth fat(default:0.0)
```

```

#ALL_LINEWD_FAT = 0.0
# Lower 200DPI linewidth fat(use FAX) (default:0.0)
#200DPI_LINEWD_FAT = 0.0
#symbol_line_width(unit:mm) (0.0 --> 20.0mm)
SYMBOL_LINEWIDTH = 0.0
# output size (unit:mm) (0.01 --> 1189.0*6)
#OUTPUT_SIZE_X = 1000.0
#OUTPUT_SIZE_Y = 596.0
# output resolution (unit:DPI) (72 --> 600)
OUT_RESOLUTION = 400
# raster working memory (unit:Mbytes)
BUNDLE_MEMORY = 64.0
#BUNDLE_MEMORY = 32.0
# output device scale (0.9 --> 1.1)
DEVICE_SCALE = 1.0
# auto offset & auto paper_size margin(unit:mm) : used -g option
AUTO_OFFSET_X = 5.0
AUTO_OFFSET_Y = 5.0
# filling mode on(1)/off(0)
FILL_MODE = 1
# paper margin (unit:mm) (default:0.0) (0.0 --> 999.99)
PAPER_MARGIN_X = 5.0
PAPER_MARGIN_Y = 5.0
# dashline cap (0:cut , 1:circle(default) , 2:square)
#DASHLINE_CAP = 0
# TIFF byte order (default:MPU depend , motorola:M , intel:I)
#TIFF_BYTEORDER=I
#TIFF_BYTEORDER=M
# if -a(-aj) on , space_area (0:no cut(default) , 1:cut)
#CUT_SPACE_AREA = 1
# output paper_size overflow limits(unit:mm) (0-->100(default:5)) (-aj option on)
PAPER_X_LIMITS = 10.0
PAPER_Y_LIMITS = 10.0
# dashline pattern down/up ratio(10%-->900%)
#DASH_DOWN_RATIO = 100
#DASH_UP_RATIO = 100
# color rgb table (red,green,blue) (0-->1000)
# black:0,0,0 red:1000,0,0 white:1000,1000,1000
#COLORRGB=(C0=0,0,0,C1=0,0,0,C2=1000,0,0,C3=1000,1000,1000,C4=0,0,0,C5=1000,0,0)
#COLORRGB=(C6=0,0,0,C7=1000,0,0,C8=1000,1000,1000,C9=0,0,0,C10=1000,0,0)
# merge fill logic(0:OR(default),1:REPLACE,2:XOR,3:AND)
#MERGE_LOGIC = 0
# hatch pattern bit(0:8x8,1:16x16,2:32x32(default))
#HATCH_PATBIT = 2
# output area size expand(only -g) (0.0-->100.0mm) (default:0.0)
#OUTAREA_UP_X = 0.5
#OUTAREA_UP_Y = 0.5
# 1dot control(0:1_dot ,1:depend on line_width(default))
#SAME_PT_DOT = 0
# A_series paper out(-g option) (Y:yes,N:no(input_size) (Default))
#A_PAPER_OUT = Y
# width byte boundary(0:off(default),byte_bound(1,2,4))
BYTE_BOUNDARY_X = 1
# height byte boundary(0:off(default),byte_bound(1,2,4))
#BYTE_BOUNDARY_Y = 1
##### LONG_PAPER RASTER #####
# paper_scale (0:off(default),1:on(-aj)&clip off,2:on(-aj)&clip on
# 10:on(not adjust scale(-aj off))&clip on)
# 11:on(not adjust scale(-aj off))&clip on&scale=1.0)
# 12:on(not adjust scale(-aj off))&clip off)
# 13:on(not adjust scale(-aj off))&clip off&scale=1.0)
#PAPER_SCALE = 11
# paper_size (unit:mm) (A0 --> A4, A0L --> A4L)
# A0(1189,841),A1(841,594),A2(594,420),A3(420,297),A4(297,210)
A0_OUTSIZE_X = 1189
A0_OUTSIZE_Y = 841
A1_OUTSIZE_X = 841
A1_OUTSIZE_Y = 594
A2_OUTSIZE_X = 594
A2_OUTSIZE_Y = 420
A3_OUTSIZE_X = 420

```

```
A3_OUTSIZE_Y = 297
A4_OUTSIZE_X = 297
A4_OUTSIZE_Y = 210
A0L_OUTSIZE_X = 7134
A0L_OUTSIZE_Y = 841
A1L_OUTSIZE_X = 5945
A1L_OUTSIZE_Y = 594
A2L_OUTSIZE_X = 4756
A2L_OUTSIZE_Y = 420
A3L_OUTSIZE_X = 3567
A3L_OUTSIZE_Y = 297
A4L_OUTSIZE_X = 2378
A4L_OUTSIZE_Y = 210
# paper_margin (unit:mm(default:0)) (A0 --> A4, A0L --> A4L)
A0_PAPER_MARGIN = 5
A1_PAPER_MARGIN = 5
A2_PAPER_MARGIN = 5
A3_PAPER_MARGIN = 5
A4_PAPER_MARGIN = 5
A0L_PAPER_MARGIN = 5
A1L_PAPER_MARGIN = 5
A2L_PAPER_MARGIN = 5
A3L_PAPER_MARGIN = 5
A4L_PAPER_MARGIN = 5
# paper_rotation(unit:degree(default:0)) (0, 90, 180, 270 (CCW))
A0_ROTATE = 0
A1_ROTATE = 0
A2_ROTATE = 0
A3_ROTATE = 0
A4_ROTATE = 0
A0L_ROTATE = 0
A1L_ROTATE = 0
A2L_ROTATE = 0
A3L_ROTATE = 0
A4L_ROTATE = 0
#####end rasl_out.atr#####
```

- ・ 出力属性データ変換属性パラメータファイルのサンプル(カラー(H23RSC)用) (rascl_out.atr)

```
#####rascl_out.atr#####
# output viersion (TIFF, NO_COMPRESS, SUN_RASTER, XWD, BMP, PNG, JPEG)
#OUT_VERSION=TIFF_NON
OUT_VERSION=TIFF_PACK
#OUT_VERSION=SUN_RASTER
#OUT_VERSION=NO_COMPRESS
#OUT_VERSION=XWD
#OUT_VERSION=BMP
#OUT_VERSION=PNG
#OUT_VERSION=JPEG
#OUT_VERSION=HPRTL_ROW
#OUT_VERSION=HPRTL_NON
#OUT_VERSION=HPRTL_PACK
# JPEG quality (0:worst --> 75:default --> 100:best)
#JPEG_QUALITY = 75
# JPEG progressive (Y/N) (default:N)
#JPEG_PROG = N
#symbol height_scale (0.01 --> 99.99)
OUT_SYMHT=1.0
# symbol width_scale (0.01 --> 99.99)
OUT_SYMWD=1.0
# symbol space (unit:mm) (0.0 --> 99.99)
OUT_SYMSP=0.0
# line type pattern set(pattern_length * scale_factor) (unit:mm)
#DOT_LINE = 1.0, 1.0
#DASH_LINE = 2.0, 1.0
#CENTER_LINE = 10.0, 1.0, 1.0
#DIVIDE_LINE = 10.0, 1.0, 1.0
# color/width cross conversion table (max 256 tables)
COLORWIDTH=(C0=0.1, C1=0.1, C2=0.2, C3=0.3, C4=0.4, C5=0.5, C6=0.3, C7=0.7, C8=0.8)
#COLORWIDTH=(C1=0.01, C2=0.01, C3=0.01, C4=0.01, C5=0.01, C6=0.01, C7=0.01, C8=0.01)
#COLORWIDTH=(C9=0.1, C10=0.1, C11=0.1, C12=0.2, C13=0.3, C14=0.4, C15=0.5, C16=0.6)
#COLORWIDTH=(C1=0.1, C2=0.1, C3=0.1, C4=0.1, C5=0.35, C6=0.35, C7=0.35, C8=0.35)
#COLORWIDTH=(C1=0.01, C2=0.01, C3=0.01, C4=0.01, C5=0.01, C6=0.01, C7=0.01, C8=0.01)
#COLORWIDTH=(C128=0.01, C254=0.01, C255=0.01)
# all vector (without symbol) linewidth fat(default:0.0)
#ALL_LINEWD_FAT = 0.0
# Lower 200DPI linewidth fat(use FAX) (default:0.0)
#200DPI_LINEWD_FAT = 0.0
#symbol line_width(unit:mm) (0.0 --> 20.0mm)
SYMBOL_LINEWIDTH = 0.01
# output size (unit:mm) (0.01 --> 1189.0*6)
#OUTPUT_SIZE_X = 1000.0
#OUTPUT_SIZE_Y = 596.0
# output resolution (unit:DPI) (72 --> 600)
OUT_RESOLUTION = 400
# raster working memory (unit:Mbytes)
BUNDLE_MEMORY = 64.0
#BUNDLE_MEMORY = 10.0
# output device scale (0.9 --> 1.1)
DEVICE_SCALE = 1.0
# auto offset & auto paper_size margin(unit:mm) : used -g option
AUTO_OFFSET_X = 5.0
AUTO_OFFSET_Y = 5.0
# filling mode on(1)/off(0)
FILL_MODE = 1
# paper margin (unit:mm) (default:0.0) (0.0 --> 999.99)
PAPER_MARGIN_X = 5.0
PAPER_MARGIN_Y = 5.0
# dashline cap (0:cut , 1:circle(default) , 2:square)
#DASHLINE_CAP = 0
# TIFF byte order (default:MPU depend , motorola:M , intel:I)
#TIFF_BYTEORDER=I
#TIFF_BYTEORDER=M
# if -a(-aj) on , space_area (0:no cut(default) , 1:cut)
#CUT_SPACE_AREA = 1
# output paper_size overflow limits(unit:mm) (0-->100(default:5)) (-aj option on)
```

```

PAPER_X_LIMITS = 10.0
PAPER_Y_LIMITS = 10.0
# dashline pattern down/up ratio(10%-->900%)
#DASH_DOWN_RATIO = 100
#DASH_UP_RATIO = 100
# color rgb table (red,green,blue) (0-->1000)
# black:0,0,0 red:1000,0,0 white:1000,1000,1000
COLORRGB=(C0=0,0,0,C1=0,0,0,C2=1000,0,0,C3=1000,1000,1000,C4=0,0,0,C5=1000,0,0)
COLORRGB=(C6=0,0,0,C7=1000,0,0,C8=1000,1000,1000,C9=0,0,0,C10=1000,0,0)
# merge fill logic(0:OR(default),1:REPLACE,2:XOR,3:AND)
#MERGE_LOGIC = 0
# hatch pattern bit(0:8x8,1:16x16,2:32x32(default))
#HATCH_PATBIT = 2
# output area size expand(only -g) (0.0-->100.0mm) (default:0.0)
#OUTAREA_UP_X = 0.5
#OUTAREA_UP_Y = 0.5
# 1dot control(0:1_dot,1:depend on line_width(default))
#SAME_PT_DOT = 0
# A_series paper out(-g option) (Y:yes,N:no(input_size) (Default))
#A_PAPER_OUT = Y
# width byte boundary(0:off(default),byte_bound(1,2,4))
BYTE_BOUNDARY_X = 1
# height byte boundary(0:off(default),byte_bound(1,2,4))
#BYTE_BOUNDARY_Y = 1
##### LONG_PAPER RASTER #####
# paper_scale (0:off(default),1:on(-aj)&clip off,2:on(-aj)&clip on
#          10:on(not adjust scale(-aj off))&clip on
#          11:on(not adjust scale(-aj off))&clip on&scale=1.0)
#          12:on(not adjust scale(-aj off))&clip off)
#          13:on(not adjust scale(-aj off))&clip off&scale=1.0)
#PAPER_SCALE = 10
#PAPER_SCALE = 2
#PAPER_SCALE = 1
# paper_size (unit:mm) (A0 --> A4, A0L --> A4L)
# A0(1189,841),A1(841,594),A2(594,420),A3(420,297),A4(297,210)
AO_OUTSIZE_X = 1189
AO_OUTSIZE_Y = 841
A1_OUTSIZE_X = 841
A1_OUTSIZE_Y = 594
A2_OUTSIZE_X = 594
A2_OUTSIZE_Y = 420
A3_OUTSIZE_X = 420
A3_OUTSIZE_Y = 297
#A3_OUTSIZE_X = 594
#A3_OUTSIZE_Y = 420
#A3_OUTSIZE_X = 297
#A3_OUTSIZE_Y = 210
A4_OUTSIZE_X = 297
A4_OUTSIZE_Y = 210
A0L_OUTSIZE_X = 7134
#A0L_OUTSIZE_X = 2378
#A0L_OUTSIZE_X = 1682
A0L_OUTSIZE_Y = 841
A1L_OUTSIZE_X = 5945
A1L_OUTSIZE_Y = 594
A2L_OUTSIZE_X = 4756
A2L_OUTSIZE_Y = 420
A3L_OUTSIZE_X = 3567
A3L_OUTSIZE_Y = 297
A4L_OUTSIZE_X = 2378
A4L_OUTSIZE_Y = 210
# paper_margin (unit:mm(default:0)) (A0 --> A4, A0L --> A4L)
AO_PAPER_MARGIN = 5
A1_PAPER_MARGIN = 5
A2_PAPER_MARGIN = 5
A3_PAPER_MARGIN = 5
A4_PAPER_MARGIN = 5
A0L_PAPER_MARGIN = 5
A1L_PAPER_MARGIN = 5
A2L_PAPER_MARGIN = 5
A3L_PAPER_MARGIN = 5

```

```
A4L_PAPER_MARGIN = 5
# paper_rotation(unit:degree(default:0)) (0, 90, 180, 270(CCW))
A0_ROTATE = 0
A1_ROTATE = 0
A2_ROTATE = 0
A3_ROTATE = 0
A4_ROTATE = 0
A0L_ROTATE = 0
A1L_ROTATE = 0
A2L_ROTATE = 0
A3L_ROTATE = 0
A4L_ROTATE = 0
# color yusen
# group_no. (1-16), group_item(color_no:1-256,T:text,*:others)
# Available HP-GL
#COLOR_YUSEN = 1, 1-256
#COLOR_YUSEN = 1, 12-16
#COLOR_YUSEN = 1, *
#COLOR_YUSEN = 2, 1
#COLOR_YUSEN = 3, 3
#COLOR_YUSEN = 4, 11
#COLOR_YUSEN = 5, 5
#COLOR_YUSEN = 6, T
#####end rascl_out. atr#####
```

サンプル例

※実行ファイルに H23RS を使用した操作例となります。

例： sample.pltと同じ大きさになるように sample.tifへ変換
原点移動を行い入力データの最小値を第一象限の(0,0)へ移動します。

```
C:\>H23RS sample.plt -Osample.tif -AG -I -Uhp2_in.atr -Zrasl_out.atr  
Copyright(c) 1992-2011 by Kernel Computer System Co.,Ltd. All rights reserved.
```

HP-GL/2(sample.plt) -> RASTER(sample.tif) ファイル コンバータ

ただいま、ファイル変換中です。

入力データの範囲 (単位:mm)
Xmin=xxx Ymin=xxx Xmax=xxx Ymax=xxx

出力データの範囲 (単位:mm)
Xmin=xxx Ymin=xxx Xmax=xxx Ymax=xxx

ベクター/ラスター変換処理 100%終了

H23RS変換終了

※-AGを指定すると実データのみ(余白無視)で変換されますので、変換後のファイルを印刷した場合には、切れてしまう場合があります。余白を作成するには、出力属性ファイル(rasl_out.atr)のAUTO_OFFSET_X(AUTO_OFFSET_Y)で余白の設定し変換します。

例： sample.pltの図面サイズから自動で用紙サイズ(A系列)を求めてセンタリング(中央配置)を行い sample.tifへ変換します。

```
C:\>H23RS sample.plt -Osample.tif -AL -I -Uhp2_in.atr -Zrasl_out.atr  
Copyright(c) 1992-2011 by Kernel Computer System Co.,Ltd. All rights reserved.
```

HP-GL/2(sample.plt) -> RASTER(sample.tif) ファイル コンバータ

ただいま、ファイル変換中です。

入力データの範囲 (単位:mm)
Xmin=xxx Ymin=xxx Xmax=xxx Ymax=xxx

出力データの範囲 (単位:mm)
Xmin=xxx Ymin=xxx Xmax=xxx Ymax=xxx

ベクター/ラスター変換処理 100%終了

H23RS変換終了

例： sample.pltを90度回転して sample.tifへ変換。-AGで第一象限に原点移動。

```
C:\>H23RS sample.plt -Osample.tif -R90 -I -AG -Uhp2_in.atr -Zrasl_out.atr  
Copyright(c) 1992-2011 by Kernel Computer System Co.,Ltd. All rights reserved.
```

HP-GL/2(sample.plt) -> RASTER(sample.tif) ファイル コンバータ

ただいま、ファイル変換中です。

入力データの範囲 (単位:mm)
Xmin=xxx Ymin=xxx Xmax=xxx Ymax=xxx

出力データの範囲 (単位:mm)
Xmin=xxx Ymin=xxx Xmax=xxx Ymax=xxx

ベクター/ラスター変換処理 100%終了

H23RS変換終了

例：複数ページファイル(sample.plt)を1ページ毎にTIFFファイル(sample.tif)へ変換

```
C:\¥>H23RS sample.plt -Osample.tif -AG -MFILE -I -Uhp2_in.atr -Zrasl_out.atr  
Copyright(c) 1992-2011 by Kernel Computer System Co.,Ltd. All rights reserved.
```

HP-GL/2(sample.plt) -> RASTER(sample.tif) ファイル コンバータ

ただいま、ファイル変換中です。

入力データの範囲 (単位:mm)
Xmin=xxx Ymin=xxx Xmax=xxx Ymax=xxx

出力データの範囲 (単位:mm)
Xmin=xxx Ymin=xxx Xmax=xxx Ymax=xxx

ベクター/ラスター変換処理 100%終了

H23RS変換終了

※ 変換された TIFF ファイルのファイル名は、指定された出力ファイル名にページ番号が
付加されて変換されます。

