File created: 20-Jul-88 17:47:02 **{MCS:MCS:STANFORD}<LANE>HPGL.;24**

changes to: (FNS \DRAWLINE.HPGL \FONT.HPGL \INIT.HPGL HARDCOPYW.HPGL)

previous date: 20-Jul-88 17:34:42 {MCS:MCS:STANFORD}<LANE>HPGL.;23

Read Table: INTERLISP

Package: INTERLISP

Format: XCCS

;;


```
(RPAQQ HPGLCOMS
       ((* * User Functions)
        (FNS MAKEHPGL OPENHPGLSTREAM HARDCOPYW.HPGL)
        (* * ImageOp Functions)
        (FNS \BITBLT.HPGL \BLTSHADE.HPGL \CLOSEFN.HPGL \COLOR.HPGL \DRAWARC.HPGL \DRAWCIRCLE.HPGL
             \DRAWCURVE.HPGL \DRAWLINE.HPGL \DRAWPOLYGON.HPGL \FILLCIRCLE.HPGL \FONT.HPGL \LEFTMARGIN.HPGL
             \LINEFEED.HPGL \MOVETO.HPGL \RESET.HPGL \RIGHTMARGIN.HPGL \ROTATE.HPGL \SCALEDBITBLT.HPGL
             \STRINGWIDTH.HPGL \CLIPPINGREGION.HPGL \TERPRI.HPGL \XPOSITION.HPGL \YPOSITION.HPGL)
        (* * Internal Functions)
        (FNS \DUMPSTRING.HPGL \FONTCREATE.HPGL \INIT.HPGL \OUTCHAR.HPGL \SEARCH.HPGL.FONTS \FILL.HPGL
             \DASHING.HPGL)
        (* * etc.)
        (VARS HPGL.FONTS HPGL.OPTIONS HPGL.FONT.EXPANSIONS HPGL.DASHING (SKETCHINCOLORFLG T))
        (INITVARS (HPGL.TERMINATOR (CHARACTER (CHARCODE ;)))
               (HPGL.SEPARATOR (CHARACTER (CHARCODE %,)))
               (HPGL.TEXT.TERMINATOR (CHARACTER (CHARCODE ^A)))
               HPGL.CHORD.ANGLE HPGL.PATTERN.LENGTH \HPGLIMAGEOPS \NULLFDEV SK.DASHING.PATTERNS)
        (GLOBALVARS HPGL.FONTS HPGL.OPTIONS HPGL.FONT.EXPANSIONS HPGL.DASHING HPGL.TERMINATOR HPGL.SEPARATOR
               HPGL.TEXT.TERMINATOR HPGL.CHORD.ANGLE HPGL.PATTERN.LENGTH \HPGLIMAGEOPS \NULLFDEV)
        (DECLARE%: DOEVAL@COMPILE DONTCOPY (FILES UTILISOPRS)
               (ALISTS (PRINTOUTMACROS !, !; !!;))
               (RECORDS PLOTTERDATA))
        [ADDVARS (PRINTERTYPES ((PLOTTER HPGL)
                                (CANPRINT (HPGL))
                                (STATUS TRUE)
                                (BITMAPFILE (HARDCOPYW.HPGL FILE BITMAP SCALEFACTOR REGION ROTATION TITLE))
                                (PROPERTIES NILL)))
               [PRINTFILETYPES (HPGL (EXTENSION (HPGL PLOT))
                                     (CONVERSION (TEXT MAKEHPGL TEDIT (LAMBDA (FILE PFILE)
                                                                              (SETQ FILE (OPENTEXTSTREAM FILE))
                                                                              (TEDIT.FORMAT.HARDCOPY FILE PFILE T
                                                                                     NIL NIL NIL 'HPGL)
                                                                              (CLOSEF? FILE)
                                                                              PFILE]
               (IMAGESTREAMTYPES (HPGL (OPENSTREAM OPENHPGLSTREAM)
                                       (FONTCREATE \FONTCREATE.HPGL)
                                       (FONTSAVAILABLE \SEARCH.HPGL.FONTS)
                                       (CREATECHARSET NILL]
        (P [if (FGETD (FUNCTION SK.DASHING.LABEL))
               then
               (for ENTRY in HPGL.DASHING do (push SK.DASHING.PATTERNS (LIST (SK.DASHING.LABEL (CDR ENTRY))
                                                                             (CDR ENTRY]
          (\INIT.HPGL))))
```

(* * User Functions)

```
(DEFINEQ
```

(**MAKEHPGL**

```
  [LAMBDA (FILE PFILE FONTS HEADING TABS)                    (* cdl "12-Jun-85 11:22")
    (TEXTTOIMAGEFILE FILE PFILE 'HPGL FONTS HEADING TABS])
```

(**OPENHPGLSTREAM**

```
  [LAMBDA (FILE OPTIONS)                                     ; Edited  8-Sep-87 08:50 by cdl
                                                             (* DECLARATIONS%: (RECORD PAIR
                                                             (KEY VALUE)))
    (LET (HPGLSTREAM POSITION (STREAM (OPENSTREAM FILE 'OUTPUT))
                (SCALE (create POSITION
                               XCOORD _ SCREENWIDTH
                               YCOORD _ SCREENHEIGHT)))
         (if (AND (SETQ POSITION (LISTGET OPTIONS 'SCALE))
                  (POSITIONP POSITION))
             then (SETQ SCALE POSITION))
         (SETQ HPGLSTREAM (create STREAM
                                  IMAGEOPS _ \HPGLIMAGEOPS
                                  IMAGEDATA _ (create PLOTTERDATA
                                                      PD.STREAM _ STREAM
                                                      PD.SCALE _ SCALE
                                                      PD.RIGHTMARGIN _ (with POSITION SCALE XCOORD))
```

```
                                OUTCHARFN _ (FUNCTION \OUTCHAR.HPGL)
                                CBUFPTR _ NIL
                                CBUFSIZE _ 0
                                DEVICE _ \NULLFDEV using STREAM))
          (with STREAM STREAM (SETQ LINELENGTH MAX.SMALLP))
          (with POSITION SCALE (printout STREAM "DF" !; "SC" "0" !, XCOORD !, "0" !, YCOORD !; "DT" !!; !;))
          [bind ENTRY for PAIR on OPTIONS by (CDDR PAIR) do (with PAIR PAIR (if (SETQ ENTRY (ASSOC KEY HPGL.OPTIONS)
                                                                                )
                                                                               then (printout STREAM (CDR ENTRY)
                                                                                            VALUE !;]
          (DSPFONT DEFAULTFONT HPGLSTREAM)
          (DSPRESET HPGLSTREAM)
          HPGLSTREAM])
```

(**HARDCOPYW.HPGL**

```
  [LAMBDA (FILE BITMAP SCALEFACTOR REGION ROTATION TITLE)        ; Edited 20-Jul-88 17:11 by cdl
    (LET ((PFILE (OPENHPGLSTREAM FILE)))
         (with REGION REGION (BITBLT BITMAP LEFT BOTTOM PFILE NIL NIL WIDTH HEIGHT))
         (CLOSEF PFILE])

)


          (* * ImageOp Functions)

(DEFINEQ
```

(**BITBLT.HPGL**

```
  [LAMBDA (BITMAP SOURCELEFT SOURCEBOTTOM STREAM DESTINATIONLEFT DESTINATIONBOTTOM WIDTH HEIGHT SOURCETYPE
                  OPERATION TEXTURE CLIPPINGREGION CLIPPEDSOURCELEFT CLIPPEDSOURCEBOTTOM)
                                                                 ; Edited 8-Sep-87 08:41 by cdl
    (\DUMPSTRING.HPGL STREAM)
    (bind (FILESTREAM _ (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
                              PD.STREAM))
       for Y from SOURCEBOTTOM to (SUB1 (PLUS SOURCEBOTTOM HEIGHT)) as J from DESTINATIONBOTTOM
       do (bind PI (STATE _ 0) for X from SOURCELEFT to (SUB1 (PLUS SOURCELEFT WIDTH)) as I from DESTINATIONLEFT
             do (if (NEQ STATE (BITMAPBIT BITMAP X Y))
                    then (if (ZEROP (SETQ STATE (IDIFFERENCE 1 STATE)))
                             then (printout FILESTREAM "PD")
                                  (if (NEQ PI (SUB1 I))
                                      then (printout FILESTREAM (SUB1 I)
                                                     !, J))
                                  (printout FILESTREAM !;)
                           else (printout FILESTREAM "PU" I !, J !;))
                  (SETQ PI I))
             finally (if (NOT (ZEROP STATE))
                         then (printout FILESTREAM "PD")
                              (if (NEQ PI (SUB1 I))
                                  then (printout FILESTREAM (SUB1 I)
                                                 !, J))
                              (printout FILESTREAM !;)))
       finally (printout FILESTREAM "PU" !;))
    T])
```

(**BLTSHADE.HPGL**

```
  [LAMBDA (TEXTURE STREAM DESTINATIONLEFT DESTINATIONBOTTOM WIDTH HEIGHT OPERATION CLIPPINGREGION)
                                                                 ; Edited 10-Nov-87 15:37 by cdl
    (SUB1VAR WIDTH)
    (SUB1VAR HEIGHT)
    (if (AND (OR (ZEROP WIDTH)
                 (ZEROP HEIGHT))
             (EQ TEXTURE BLACKSHADE))
        then                                                     (* Get around bug in plotter hardware triggered by SKETCH
                                                                 boxes)
             (DRAWLINE DESTINATIONLEFT DESTINATIONBOTTOM (PLUS DESTINATIONLEFT WIDTH)
                    (PLUS DESTINATIONBOTTOM HEIGHT)
                    NIL OPERATION STREAM)
      else (IMAGEOP 'IMMOVETO STREAM STREAM DESTINATIONLEFT DESTINATIONBOTTOM)
           (\FILL.HPGL STREAM TEXTURE)
           (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
                 (printout PD.STREAM "RA" (PLUS DESTINATIONLEFT WIDTH)
                        !,
                        (PLUS DESTINATIONBOTTOM HEIGHT)
                        !;)))
    T])
```

(**CLOSEFN.HPGL**

```
  [LAMBDA (STREAM)                                               ; Edited 8-Sep-87 08:34 by cdl
    (\DUMPSTRING.HPGL STREAM)
    (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
          (printout PD.STREAM "PU" !;)
          (CLOSEF? PD.STREAM)
          (SETQ PD.STREAM NIL))
    T])
```

(\COLOR.HPGL

```
[LAMBDA (STREAM COLOR)                                          ; Edited  8-Dec-87 17:10 by cdl
                                                                (* DECLARATIONS%: (RECORD ENTRY
                                                                (NAME . VALUES)))
    (DECLARE (GLOBALVARS COLORNAMES))
    (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
          (PROG1 PD.COLOR
                 (if COLOR
                     then [if (LITATOM COLOR)
                              then (SETQ COLOR (for ENTRY in COLORNAMES as I from 1
                                                  thereis (with ENTRY ENTRY (EQ COLOR NAME))
                                                  yield
                                                  (DIFFERENCE (LENGTH COLORNAMES)
                                                         I)))
                            elseif (RGBP COLOR)
                              then (SETQ COLOR (for ENTRY in COLORNAMES as I from 1
                                                  thereis (with ENTRY ENTRY (EQUAL COLOR VALUES))
                                                  yield
                                                  (DIFFERENCE (LENGTH COLORNAMES)
                                                         I]
                          (if (AND (FIXP COLOR)
                                   (NEQ COLOR PD.COLOR))
                              then (\DUMPSTRING.HPGL STREAM)
                                   (printout PD.STREAM "SP" (ADD1 (SETQ PD.COLOR COLOR))
                                          !;)))))])
```

(\DRAWARC.HPGL

```
[LAMBDA (STREAM CENTERX CENTERY RADIUS STARTANGLE NDEGREES BRUSH DASHING)
                                                                ; Edited 14-Sep-87 10:57 by cdl
    (DECLARE (SPECVARS . T))
    (\DUMPSTRING.HPGL STREAM)
    [if (LISTP BRUSH)
        then (with BRUSH BRUSH (if BRUSHCOLOR
                                   then (IMAGEOP 'IMCOLOR STREAM STREAM BRUSHCOLOR]
    (RESETLST
        [RESETSAVE NIL `(\DASHING.HPGL ,STREAM ,(\DASHING.HPGL STREAM DASHING]
        [RESETSAVE NIL `(DSPCOLOR ,(IMAGEOP 'IMCOLOR STREAM STREAM (if (LISTP BRUSH)
                                                                       then (with BRUSH BRUSH BRUSHCOLOR)))
                                  ,STREAM]
        (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
              (with POSITION PD.POSITION (printout PD.STREAM "PU" (SETQ XCOORD CENTERX)
                                                 !,
                                                 (SETQ YCOORD CENTERY)
                                                 !; "EW" RADIUS !, (PLUS STARTANGLE 90)
                                                 !, NDEGREES)
                    (if HPGL.CHORD.ANGLE
                        then (printout PD.STREAM !, HPGL.CHORD.ANGLE))
                    (printout PD.STREAM !;))))
    T])
```

(\DRAWCIRCLE.HPGL

```
[LAMBDA (STREAM CENTERX CENTERY RADIUS BRUSH DASHING)           ; Edited 14-Sep-87 10:54 by cdl
    (DECLARE (SPECVARS . T))
    (\DUMPSTRING.HPGL STREAM)
    [if (LISTP BRUSH)
        then (with BRUSH BRUSH (if BRUSHCOLOR
                                   then (IMAGEOP 'IMCOLOR STREAM STREAM BRUSHCOLOR]
    (RESETLST
        [RESETSAVE NIL `(\DASHING.HPGL ,STREAM ,(\DASHING.HPGL STREAM DASHING]
        [RESETSAVE NIL `(DSPCOLOR ,(IMAGEOP 'IMCOLOR STREAM STREAM (if (LISTP BRUSH)
                                                                       then (with BRUSH BRUSH BRUSHCOLOR)))
                                  ,STREAM]
        (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
              (with POSITION PD.POSITION (printout PD.STREAM "PU" (SETQ XCOORD CENTERX)
                                                 !,
                                                 (SETQ YCOORD CENTERY)
                                                 !; "CI" RADIUS)
                    (if HPGL.CHORD.ANGLE
                        then (printout PD.STREAM !, HPGL.CHORD.ANGLE))
                    (printout PD.STREAM !;))))
    T])
```

(\DRAWCURVE.HPGL

```
[LAMBDA (STREAM KNOTS CLOSED BRUSH DASHING)                     ; Edited  8-Sep-87 11:25 by cdl
    (DECLARE (SPECVARS . T))
    (\DUMPSTRING.HPGL STREAM)
    (if (FGETD 'DRAWCURVE.STREAM)
        then (RESETLST
                 [RESETSAVE NIL `(DSPCOLOR ,(IMAGEOP 'IMCOLOR STREAM STREAM (if (LISTP BRUSH)
                                                                                then (with BRUSH BRUSH BRUSHCOLOR)
                                                                               ))
                                           ,STREAM]
```

```
                (DRAWCURVE.STREAM STREAM KNOTS CLOSED BRUSH DASHING))
      else (IMAGEOP 'IMDRAWPOLYGON STREAM STREAM KNOTS CLOSED BRUSH DASHING])
```

(\DRAWLINE.HPGL

```
(LAMBDA (STREAM X1 Y1 X2 Y2 WIDTH OPERATION COLOR DASHING)      ; Edited 20-Jul-88 17:45 by cdl
  (DECLARE (SPECVARS . T))
  [if [AND DASHING (NOT (bind (DASHING _ (MKLIST DASHING)) for ENTRY in HPGL.DASHING
                           thereis (EQUAL DASHING (CDR ENTRY]
      then                                                      (* Not a hardware dashing pattern)
           (DRAWDASHEDLINE X1 Y1 X2 Y2 (OR WIDTH 1)
                  OPERATION STREAM COLOR DASHING)
    else (\DUMPSTRING.HPGL STREAM)
         (RESETLST
             [RESETSAVE NIL `(\DASHING.HPGL ,STREAM ,(\DASHING.HPGL STREAM DASHING]
             [RESETSAVE NIL `(DSPCOLOR ,(IMAGEOP 'IMCOLOR STREAM STREAM COLOR)
                              ,STREAM]
             (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
                   (with POSITION PD.POSITION (if [NOT (AND (OR (EQ X1 T)
                                                                (EQ X1 XCOORD))
                                                            (OR (EQ Y1 T)
                                                                (EQ Y1 YCOORD]
                                                  then (printout PD.STREAM "PU" (if (EQ X1 T)
                                                                                    then XCOORD
                                                                                  else X1)
                                                                                !,
                                                                                (if (EQ Y1 T)
                                                                                    then YCOORD
                                                                                  else Y1)
                                                                                !;))
                                          (printout PD.STREAM "PD" (SETQ XCOORD X2)
                                                 !,
                                                 (SETQ YCOORD Y2)
                                                 !;)))))]
  T])
```

(\DRAWPOLYGON.HPGL

```
(LAMBDA (STREAM POINTS CLOSED BRUSH DASHING)                    ; Edited  8-Sep-87 08:22 by cdl
  (DECLARE (SPECVARS . T))
  (\DUMPSTRING.HPGL STREAM)
  (RESETLST
      [RESETSAVE NIL `(\DASHING.HPGL ,STREAM ,(\DASHING.HPGL STREAM DASHING]
      [RESETSAVE NIL `(DSPCOLOR ,(IMAGEOP 'IMCOLOR STREAM STREAM (if (LISTP BRUSH)
                                                                     then (with BRUSH BRUSH BRUSHCOLOR)))
                               ,STREAM]
      (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
            (with POSITION (CAR POINTS)
                  (printout PD.STREAM "PU" XCOORD !, YCOORD !; "PD"))
            (for POINT on (CDR POINTS) do (with POSITION (CAR POINT)
                                                 (printout PD.STREAM XCOORD !, YCOORD))
                                          (if (CDR POINT)
                                              then (printout PD.STREAM !,)))
            (if CLOSED
                then (with POSITION (CAR POINTS)
                           (printout PD.STREAM XCOORD !, YCOORD)))
            (PRINTOUT PD.STREAM !;)
            (with POSITION (CAR (LAST POINTS))
                  (create POSITION
                          XCOORD _ XCOORD
                          YCOORD _ YCOORD smashing PD.POSITION))))
  T])
```

(\FILLCIRCLE.HPGL

```
(LAMBDA (STREAM CENTERX CENTERY RADIUS TEXTURE)                 ; Edited 14-Sep-87 11:25 by cdl
  (\DUMPSTRING.HPGL STREAM)
  (\FILL.HPGL STREAM TEXTURE)
  (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
        (with POSITION PD.POSITION (printout PD.STREAM "PU" (SETQ XCOORD CENTERX)
                                           !,
                                           (SETQ YCOORD CENTERY)
                                           !; "WG" RADIUS !, "0" !, "360")
              (if HPGL.CHORD.ANGLE
                  then (printout PD.STREAM !, HPGL.CHORD.ANGLE))
              (printout PD.STREAM !;)))
  T])
```

(\FONT.HPGL

```
(LAMBDA (STREAM FONT)                                           ; Edited 20-Jul-88 17:34 by cdl
  [if (type? FONTCLASS FONT)
      then (SETQ FONT (FONTCLASSCOMPONENT FONT (IMAGESTREAMTYPE STREAM]
  (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
        (if (AND FONT (NEQ FONT PD.FONT))
            then (\DUMPSTRING.HPGL STREAM)
                 [with FONTDESCRIPTOR FONT (if (NEQ FONTFAMILY (fetch FONTFAMILY of PD.FONT))
```

```
                                        then (printout PD.STREAM "CS" (OR (CDR (FASSOC FONTFAMILY
                                                                                  HPGL.FONTS))
                                                                          (CONSTANT null))
                                                                      !;))
                                (if (NEQ ROTATION (fetch ROTATION of PD.FONT))
                                    then (printout PD.STREAM "DI")
                                         (if (AND ROTATION (NOT (ZEROP ROTATION)))
                                             then (printout PD.STREAM (COS ROTATION)
                                                            !,
                                                            (SIN ROTATION)))
                                         (printout PD.STREAM !;))
                                (with POSITION PD.SCALE (printout PD.STREAM "SR")
                                      (PRINTNUM '(FLOAT NIL 3)
                                             (QUOTIENT (QUOTIENT [TIMES FONTAVGCHARWIDTH (with FONTFACE FONTFACE
                                                                                                (CDR (ASSOC EXPANSION
                                                                                                       HPGL.FONT.EXPANSIONS
                                                                                                           ]
                                                                                3)
                                                       XCOORD)
                                             PD.STREAM)
                                      (printout PD.STREAM !,)
                                      (PRINTNUM '(FLOAT NIL 3)
                                             (QUOTIENT (TIMES \SFHeight 100.0)
                                                       YCOORD)
                                             PD.STREAM)
                                      (printout PD.STREAM !;))
                                (with FONTFACE FONTFACE (if (NEQ SLOPE (fetch (FONTFACE SLOPE) of (fetch (FONTDESCRIPTOR
                                                                                                           FONTFACE)
                                                                                                  of PD.FONT)))
                                                           then (printout PD.STREAM "SL" (SELECTQ SLOPE
                                                                                                (REGULAR (CONSTANT null))
                                                                                                (ITALIC 1)
                                                                                                (SHOULDNT))
                                                                                        !;]
                                      (PROG1 PD.FONT (SETQ PD.FONT FONT))
                                else PD.FONT])
```

(**LEFTMARGIN.HPGL**

```
[LAMBDA (STREAM XPOSITION)                                  (* cdl "25-Jun-85 15:33")
  (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
        (PROG1 PD.LEFTMARGIN
            (if XPOSITION
                then (SETQ PD.LEFTMARGIN XPOSITION)))])
```

(**LINEFEED.HPGL**

```
[LAMBDA (STREAM DELTAY)                                     (* cdl "24-Jul-85 08:01")
  (MINUS (TIMES 2 (FONTPROP (with STREAM STREAM (with PLOTTERDATA IMAGEDATA PD.FONT))
                            'HEIGHT])
```

(**MOVETO.HPGL**

```
[LAMBDA (STREAM X Y)                                        ; Edited 8-Sep-87 10:39 by cdl
  (\DUMPSTRING.HPGL STREAM)
  (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
        (with POSITION PD.POSITION (printout PD.STREAM "PU" (SETQ XCOORD X)
                                           !,
                                           (SETQ YCOORD Y)
                                           !;)))
  T])
```

(**RESET.HPGL**

```
[LAMBDA (STREAM)                                            (* cdl "19-Jul-85 16:30")
  (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
        (IMAGEOP 'IMMOVETO STREAM STREAM PD.LEFTMARGIN (PLUS (with POSITION PD.SCALE YCOORD)
                                                             (IMAGEOP 'IMLINEFEED STREAM STREAM])
```

(**RIGHTMARGIN.HPGL**

```
[LAMBDA (STREAM XPOSITION)                                  (* cdl "25-Jun-85 15:34")
  (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
        (PROG1 PD.RIGHTMARGIN
            (if XPOSITION
                then (SETQ PD.RIGHTMARGIN XPOSITION)))])
```

(**ROTATE.HPGL**

```
[LAMBDA (STREAM ROTATION)                                   ; Edited 8-Sep-87 08:37 by cdl
  (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
        (PROG1 PD.ROTATION
            (if PD.ROTATION
                then (\DUMPSTRING.HPGL STREAM)
                     (printout PD.STREAM "RO" PD.ROTATION !;)))])
```

(\SCALEDBITBLT.HPGL

```
[LAMBDA (BITMAP SOURCELEFT SOURCEBOTTOM STREAM DESTINATIONLEFT DESTINATIONBOTTOM WIDTH HEIGHT SOURCETYPE
               OPERATION TEXTURE CLIPPINGREGION CLIPPEDSOURCELEFT CLIPPEDSOURCEBOTTOM SCALE)
                                                                    ; Edited 8-Sep-87 08:43 by cdl
  (\DUMPSTRING.HPGL STREAM)
  (bind (FILESTREAM _ (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
                            PD.STREAM))
        (SOURCEWIDTH _ (SUB1 (PLUS SOURCELEFT WIDTH))) for Y from SOURCEBOTTOM
     to (SUB1 (PLUS SOURCEBOTTOM HEIGHT)) as J from DESTINATIONBOTTOM by SCALE
     do [for Z from J to (PLUS J (SUB1 SCALE))
         do (bind PI LASTPOSITION (STATE _ 0) for X from SOURCELEFT to SOURCEWIDTH as I from DESTINATIONLEFT
               by SCALE do (if (NEQ STATE (BITMAPBIT BITMAP X Y))
                               then (if (ZEROP (SETQ STATE (DIFFERENCE 1 STATE)))
                                        then (printout FILESTREAM "PD")
                                             (if (NOT (IEQP PI (SUB1 I)))
                                                 then (printout FILESTREAM (SUB1 I)
                                                              !, Z))
                                             (printout FILESTREAM !;)
                                      else (printout FILESTREAM "PU" I !, Z !;))
                                    (SETQ PI I))
               finally (if (NOT (ZEROP STATE))
                           then (printout FILESTREAM "PD")
                                (if (NOT (IEQP PI (SUB1 I)))
                                    then (printout FILESTREAM (SUB1 I)
                                                  !, Z))
                                (printout FILESTREAM !;]
     finally (printout FILESTREAM "PU" !;))
  T])
```

(\STRINGWIDTH.HPGL

```
[LAMBDA (STREAM STRING RDTBL)                                       (* cdl "29-Apr-85 14:31")
  (STRINGWIDTH STRING (DSPFONT NIL STREAM)
         RDTBL RDTBL])
```

(\CLIPPINGREGION.HPGL

```
[LAMBDA (STREAM REGION)                                             (* cdl "16-Oct-85 10:57")
  (with STREAM STREAM (with PLOTTERDATA IMAGEDATA (with POSITION PD.SCALE (CREATEREGION 0 0 XCOORD YCOORD])
```

(\TERPRI.HPGL

```
[LAMBDA (STREAM)                                                    (* cdl "24-Jul-85 09:26")
  (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
       (with POSITION PD.POSITION (IMAGEOP 'IMMOVETO STREAM STREAM PD.LEFTMARGIN (PLUS YCOORD
                                                                                       (IMAGEOP 'IMLINEFEED
                                                                                              STREAM STREAM])
```

(\XPOSITION.HPGL

```
[LAMBDA (STREAM XPOSITION)                                          ; Edited 8-Sep-87 08:32 by cdl
  (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
       (with POSITION PD.POSITION (PROG1 XCOORD
                                         (if XPOSITION
                                             then (\DUMPSTRING.HPGL STREAM)
                                                  (printout PD.STREAM "PU" (SETQ XCOORD XPOSITION)
                                                         !, YCOORD !;)))])
```

(\YPOSITION.HPGL

```
[LAMBDA (STREAM YPOSITION)                                          ; Edited 8-Sep-87 08:31 by cdl
  (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
       (with POSITION PD.POSITION (PROG1 YCOORD
                                         (if YPOSITION
                                             then (\DUMPSTRING.HPGL STREAM)
                                                  (printout PD.STREAM "PU" XCOORD !, (SETQ YCOORD YPOSITION)
                                                         !;)))])

)


          (* * Internal Functions)

(DEFINEQ
```

(\DUMPSTRING.HPGL

```
[LAMBDA (STREAM)                                                    ; Edited 8-Sep-87 08:51 by cdl
  (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
       (if PD.TEXT
           then (printout PD.STREAM "LB")
                (for CHARCODE in (DREVERSE PD.TEXT) do (BOUT PD.STREAM CHARCODE))
                (printout PD.STREAM !!;)
                (SETQ PD.TEXT NIL)
                T])
```

(\FONTCREATE.HPGL

```
[LAMBDA (FAMILY SIZE FACE ROTATION)                          ; Edited  4-Sep-87 15:13 by cdl
  (if (ASSOC FAMILY HPGL.FONTS)
      then (LET ((WIDTHSBLOCK (\CREATECSINFOELEMENT))
                 (FONTDESCRIPTOR (create FONTDESCRIPTOR
                                         FONTDEVICE _ 'HPGL
                                         FONTFAMILY _ FAMILY
                                         FONTSIZE _ SIZE
                                         FONTFACE _ FACE
                                         ROTATION _ ROTATION
                                         \SFHeight _ SIZE
                                         \SFAscent _ SIZE
                                         \SFDescent _ 0)))
                 (bind (WIDTH _ (FIX (QUOTIENT (TIMES 3 SIZE)
                                               4)))
                    for N from 0 to 254 do (\FSETWIDTH WIDTHSBLOCK N WIDTH))
                 (with FONTDESCRIPTOR FONTDESCRIPTOR
                       (\SETCHARSETINFO FONTCHARSETVECTOR 0
                              (create CHARSETINFO
                                      WIDTHS _ WIDTHSBLOCK
                                      IMAGEWIDTHS _ WIDTHSBLOCK
                                      CHARSETASCENT _ SIZE
                                      CHARSETDESCENT _ 0)))
                 FONTDESCRIPTOR)
    else (FONTCREATE (CAAR HPGL.FONTS)
                     SIZE FACE ROTATION 'HPGL])
```

(\INIT.HPGL

```
[LAMBDA NIL                                                  ; Edited 20-Jul-88 17:04 by cdl
                                                             (* DECLARATIONS%: (RECORD CLASS
                                                             (FONTCLASSNAME PRETTYFONT# DISPLAYFD PRESSFD
                                                             INTERPRESSFD . OTHERFDS)))
  (DECLARE (GLOBALVARS FONTDEFS FONTNAME))
  (SETQ \NULLFDEV (create FDEV
                          CLOSEFILE _ (FUNCTION NILL)))
  (SETQ \HPGLIMAGEOPS (create IMAGEOPS
                              IMAGETYPE _ 'HPGL
                              IMCLOSEFN _ (FUNCTION \CLOSEFN.HPGL)
                              IMXPOSITION _ (FUNCTION \XPOSITION.HPGL)
                              IMYPOSITION _ (FUNCTION \YPOSITION.HPGL)
                              IMFONT _ (FUNCTION \FONT.HPGL)
                              IMLEFTMARGIN _ (FUNCTION \LEFTMARGIN.HPGL)
                              IMRIGHTMARGIN _ (FUNCTION \RIGHTMARGIN.HPGL)
                              IMLINEFEED _ (FUNCTION \LINEFEED.HPGL)
                              IMDRAWLINE _ (FUNCTION \DRAWLINE.HPGL)
                              IMDRAWCURVE _ (FUNCTION \DRAWCURVE.HPGL)
                              IMDRAWCIRCLE _ (FUNCTION \DRAWCIRCLE.HPGL)
                              IMDRAWELLIPSE _ (FUNCTION DRAWELLIPSEWITHDRAWCURVE)
                              IMFILLCIRCLE _ (FUNCTION \FILLCIRCLE.HPGL)
                              IMBITBLT _ (FUNCTION \BITBLT.HPGL)
                              IMBLTSHADE _ (FUNCTION \BLTSHADE.HPGL)
                              IMMOVETO _ (FUNCTION \MOVETO.HPGL)
                              IMSCALE _ [FUNCTION (LAMBDA (STREAM SCALE)
                                                    1]
                              IMTERPRI _ (FUNCTION \TERPRI.HPGL)
                              IMFONTCREATE _ 'HPGL
                              IMCOLOR _ (FUNCTION \COLOR.HPGL)
                              IMSTRINGWIDTH _ (FUNCTION \STRINGWIDTH.HPGL)
                              IMCHARWIDTH _ (FUNCTION \STRINGWIDTH.HPGL)
                              IMRESET _ (FUNCTION \RESET.HPGL)
                              IMCLIPPINGREGION _ (FUNCTION \CLIPPINGREGION.HPGL)
                              IMDRAWPOLYGON _ (FUNCTION \DRAWPOLYGON.HPGL)
                              IMSCALEDBITBLT _ (FUNCTION \SCALEDBITBLT.HPGL)
                              IMDRAWARC _ (FUNCTION \DRAWARC.HPGL)
                              IMROTATE _ (FUNCTION \ROTATE.HPGL)))
  (for FONTSET in FONTDEFS
     do [for CLASS in (CDR (ASSOC 'FONTPROFILE (CDR FONTSET))) unless (with CLASS CLASS
                                                                           (OR (NULL DISPLAYFD)
                                                                               (NULL INTERPRESSFD)
                                                                               (ASSOC 'HPGL OTHERFDS)))
           do (with CLASS CLASS (push OTHERFDS (LIST 'HPGL (CONS 'STANDARD (CDR (if (LISTP DISPLAYFD)
                                                                                     then DISPLAYFD
                                                                                   else (FONTUNPARSE DISPLAYFD]
     finally (FONTSET FONTNAME])
```

(\OUTCHAR.HPGL

```
[LAMBDA (STREAM CHARCODE)                                    (* cdl " 3-Oct-85 13:20")
  (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
        (SELCHARQ CHARCODE
             (EOL (\TERPRI.HPGL STREAM))
             (if (AND (GEQ CHARCODE (CHARCODE SPACE))
                      (LEQ CHARCODE (CHARCODE ~)))
                 then (with POSITION PD.POSITION (add XCOORD (CHARWIDTH CHARCODE PD.FONT)))
                      (push PD.TEXT CHARCODE])
```

(\SEARCH.HPGL.FONTS

```
  [LAMBDA (FAMILY SIZE FACE ROTATION DEVICE)                    (* cdl " 1-May-85 09:34")
    (if (EQ DEVICE 'HPGL)
        then (if (FASSOC FAMILY HPGL.FONTS)
                 then (LIST (LIST FAMILY SIZE FACE ROTATION DEVICE])
```

(\FILL.HPGL

```
  [LAMBDA (STREAM TEXTURE)                                      ; Edited  8-Dec-87 16:56 by cdl
                                                                (* DECLARATIONS%: (RECORD TEXTURE
                                                                (TYPE SPACING ANGLE)) (RECORD TEXTURECOLORPAIR
                                                                (TEXURE COLOR)))
    (\DUMPSTRING.HPGL STREAM)
    (if (LISTP TEXTURE)
        then (SETQ TEXTURE (with TEXTURECOLORPAIR TEXTURE (if (RGBP COLOR)
                                                              then (IMAGEOP 'IMCOLOR STREAM STREAM COLOR))
                                 TEXTURE)))
    [if (FIXP TEXTURE)
        then (SETQ TEXTURE (create TEXTURE
                                   TYPE _ (if (IEQP TEXTURE BLACKSHADE)
                                              then 1
                                            elseif (IEQP TEXTURE WHITESHADE)
                                              then 3
                                            else 4)
                                   SPACING _ 0
                                   ANGLE _ (TIMES (LOGAND TEXTURE 3)
                                                  45]
    (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
          (with TEXTURE TEXTURE (printout PD.STREAM "FT" (if (AND (FIXP TYPE)
                                                                  (GEQ TYPE 1)
                                                                  (LEQ TYPE 4))
                                                             then TYPE
                                                           else 1)
                                        !,
                                        (if (FIXP SPACING)
                                            then SPACING
                                          else 0)
                                        !,
                                        (if (AND (FIXP ANGLE)
                                                 (ZEROP (IMOD ANGLE 45)))
                                            then ANGLE
                                          else 0)
                                        !;)))
    T])
```

(\DASHING.HPGL

```
  [LAMBDA (STREAM DASHING)                                      ; Edited 14-Sep-87 11:28 by cdl
                                                                (* DECLARATIONS%: (RECORD ENTRY
                                                                (INDEX . LENGTHS)))
    (with PLOTTERDATA (with STREAM STREAM IMAGEDATA)
          (PROG1 (if PD.DASHING
                     then (CDR (ASSOC PD.DASHING HPGL.DASHING)))
                 (if DASHING
                     then (LET (INDEX)
                               [if (SETQ DASHING (MKLIST DASHING))
                                   then (SETQ INDEX (for ENTRY in HPGL.DASHING thereis (with ENTRY ENTRY
                                                                                             (EQUAL DASHING LENGTHS))
                                                                                     yield
                                                                                      (with ENTRY ENTRY INDEX]
                               (if (AND (FIXP INDEX)
                                        (NEQ INDEX PD.DASHING))
                                   then (\DUMPSTRING.HPGL STREAM)
                                        (printout PD.STREAM "LT" (SETQ PD.DASHING INDEX))
                                        (if HPGL.PATTERN.LENGTH
                                            then (printout PD.STREAM !, HPGL.PATTERN.LENGTH))
                                        (printout PD.STREAM !;)))
                   elseif PD.DASHING
                     then (\DUMPSTRING.HPGL STREAM)
                          (printout PD.STREAM "LT" !;)
                          (SETQ PD.DASHING NIL)))])

)


          (* * etc.)

(RPAQQ HPGL.FONTS
       ((STANDARD . 0)
        (9825 . 1)
        (FRENCH . 2)
        (SCANDINAVIAN . 3)
        (SPANISH . 4)
        (JISASCII . 6)
        (ROMAN . 7)
        (KATAKANA . 8)
        (IRV . 9)
```

```
        (SWEDISH . 30)
        (SWEDISH2 . 31)
        (NORWAY . 32)
        (GERMAN . 33)
        (FRENCH2 . 34)
        (BRITISH . 35)
        (ITALIAN . 36)
        (SPANISH2 . 37)
        (PORTUGUESE . 38)
        (NORWAY2 . 39)))

(RPAQQ HPGL.OPTIONS ((ROTATE . "RO")
                     (VELOCITY . "VS")
                     (PAPER . "PS")
                     (TERMINATOR . "DT")))

(RPAQQ HPGL.FONT.EXPANSIONS ((REGULAR . 200.0)
                             (COMPRESSED . 100.0)
                             (EXPANDED . 400.0)))

(RPAQQ HPGL.DASHING
       ((1 1 49)
        (2 25)
        (3 35 15)
        (4 39 5 1 5)
        (5 35 5 5 5)
        (6 25 5 5 5 5 5)))

(RPAQQ SKETCHINCOLORFLG T)

(RPAQ? HPGL.TERMINATOR (CHARACTER (CHARCODE ;)))

(RPAQ? HPGL.SEPARATOR (CHARACTER (CHARCODE %,)))

(RPAQ? HPGL.TEXT.TERMINATOR (CHARACTER (CHARCODE ^A)))

(RPAQ? HPGL.CHORD.ANGLE NIL)

(RPAQ? HPGL.PATTERN.LENGTH NIL)

(RPAQ? \HPGLIMAGEOPS NIL)

(RPAQ? \NULLFDEV NIL)

(RPAQ? SK.DASHING.PATTERNS NIL)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(GLOBALVARS HPGL.FONTS HPGL.OPTIONS HPGL.FONT.EXPANSIONS HPGL.DASHING HPGL.TERMINATOR HPGL.SEPARATOR
       HPGL.TEXT.TERMINATOR HPGL.CHORD.ANGLE HPGL.PATTERN.LENGTH \HPGLIMAGEOPS \NULLFDEV)
)

(DECLARE%: DOEVAL@COMPILE DONTCOPY

(FILESLOAD UTILISOPRS)

(ADDTOVAR PRINTOUTMACROS
          [!, (LAMBDA (COMS)
                      (CONS '(PRIN1 HPGL.SEPARATOR NIL)
                            (CDR COMS]
          [!; (LAMBDA (COMS)
                      (CONS '(PRIN1 HPGL.TERMINATOR NIL)
                            (CDR COMS]
          [!!; (LAMBDA (COMS)
                       (CONS '(PRIN1 HPGL.TEXT.TERMINATOR NIL)
                             (CDR COMS])

(DECLARE%: EVAL@COMPILE

(RECORD PLOTTERDATA (PD.STREAM PD.POSITION PD.FONT PD.TEXT PD.COLOR PD.SCALE PD.LEFTMARGIN PD.RIGHTMARGIN
                               PD.DASHING PD.ROTATION)
       PD.POSITION _ (create POSITION)
       PD.COLOR _ 0 PD.LEFTMARGIN _ 0 PD.ROTATION _ 0)
)
)

(ADDTOVAR PRINTERTYPES ((PLOTTER HPGL)
                        (CANPRINT (HPGL))
                        (STATUS TRUE)
                        (BITMAPFILE (HARDCOPYW.HPGL FILE BITMAP SCALEFACTOR REGION ROTATION TITLE))
                        (PROPERTIES NILL)))

(ADDTOVAR PRINTFILETYPES
          [HPGL (EXTENSION (HPGL PLOT))
                (CONVERSION (TEXT MAKEHPGL TEDIT (LAMBDA (FILE PFILE)
                                                         (SETQ FILE (OPENTEXTSTREAM FILE))
                                                         (TEDIT.FORMAT.HARDCOPY FILE PFILE T NIL NIL NIL
```

```
                                                              'HPGL)
                                                          (CLOSEF? FILE)
                                                          PFILE])

(ADDTOVAR IMAGESTREAMTYPES (HPGL (OPENSTREAM OPENHPGLSTREAM)
                                 (FONTCREATE \FONTCREATE.HPGL)
                                 (FONTSAVAILABLE \SEARCH.HPGL.FONTS)
                                 (CREATECHARSET NILL)))

[if (FGETD (FUNCTION SK.DASHING.LABEL))
    then (for ENTRY in HPGL.DASHING do (push SK.DASHING.PATTERNS (LIST (SK.DASHING.LABEL (CDR ENTRY))
                                                                       (CDR ENTRY]

(\INIT.HPGL)

(PUTPROPS HPGL COPYRIGHT ("Stanford University" 1985 1986 1987 1988))
```

FUNCTION INDEX

VARIABLE INDEX

RECORD INDEX