Tech Info Library

PostScript: Daisywheel emulator for the LaserWriter (2 of 2)

Article Created: 2 May 1986
Article Reviewed/Updated: 8 February 1995

TOPIC ---

This is part 2 of an article that describes how to build a daisywheel emulator for a LaserWriter printer. This information is provided by Adobe Customer Support. Apple Computer, Inc. is not responsible for the content of this article.

DISCUSSION --

Caution:
========
If you choose to use the PostScript code provided in this article, you assume all risks involved in making these changes. PostScript code, if not entered correctly, can place the LaserWriter into a condition requiring service.

```
%============================================================================
% Parameters and procedures defined after this point may be customized
% without harm (?) to the program.

/setlandscape?  false  def
/TopOfPage   10.5 72 mul def
/LeftMargin   .25  72 mul def
/fsize       11  def      %fontsize
/LnHt        11  def      %line leading
/MaxLineCt   66  def
/MaxCharCt   80  def
/FirstLine   TopOfPage fsize sub  def

%The following three lines define actual fonts, NOT font-finding procedures.
/BoldFont    /Courier-Bold     findfont fsize scalefont  def
/ItalicFont  /Courier-Oblique  findfont fsize scalefont  def
/RomanFont   /Courier          findfont fsize scalefont  def

/HT    %horiz tab -- typical implementation
       %tab set every 8 spaces, fixed.
  { 8  charct 8 mod  sub
    { 32 emit } repeat
  }def
```

```
%-----------------------------------------------------------------------------
%The following parameters define CmdDict and its sub-dictionaries.
%They can be customized to any extent.

/CmdDict  30 dict  def    %May need to be enlarged if you add a lot to it.
CmdDict begin             %Store the defs in CmdDict, rather than clutter
                          %DaisyEmDict with all this stuff.

 12  {FF} def
  9  {HT} def
 10  {LF CR} def    %Note:  The serial port has a "feature" at this time which
                    %forces all Linefeeds, CR-LF pairs, and CR's to appear as
                    %simple LF's to the "read" operator.

 13  {CR LF} def    %cr with auto-linefeed
 27  {ESC} def

/EscDict  10 dict def     %May need to be enlarged if you add a lot to it.
/ESC                      %Handle two-character "escape" sequences.
  { EscDict begin }def

% Rules for creating Command sub-dictionaries:
%    Sub-dictionaries of CmdDict may be nested to handle multi-character
%      command sequences of arbitrary length.  EscDict is an example, which
%      handles 2-char sequences beginning with "escape".  Note that EscDict
%      is created entirely within the dictionary (CmdDict) which may invoke it.
%    Each sub-dictionary must contain the re-definition of "emit" shown below.
%      This takes care of invoking popCmdStack when a multi-char command
%      becomes unrecognizable in the middle of the sequence.  That's all the
%      error-handling there is, folks!
%    Each leaf in any command sub-dictionary (i.e., the procedure def for
%      any command character that ends a command sequence) must end with
%      an invocation of popCmdStack.
%      That probably isn't clear, so please observe the use of "popCmdStack"
%      in the following defs, and do likewise!
%        Defs that do NOT need to popCmdStack:
%        -- words like ESC which are not "leafs" because they lead on to
%           further sub-dict nesting.
%        -- words in the root command dict, "CmdDict".


EscDict begin    %-----------------------------------------------------------

    /emit  { CmdError }def
    34  { RomanFont setfont popCmdStack }def
    33  { BoldFont  setfont popCmdStack }def
    88  { ItalicFont setfont popCmdStack }def
    89  { RomanFont setfont popCmdStack }def

end  %EscDict ---------------------------------------------------------------

%=============================================================================
end  %CmdDict
end  %DaisyEmDict    %End of program "Daisyprint".

%***************************************************************************

%Proper format for use:
%
%  Daisyprint CR
%  data data data data .....
%  ...data data ^D ^D
```

Article Change History:
08 Feb 1995 - Added PostScript caution.
Adobe Customer Support

Keywords: <None>

==