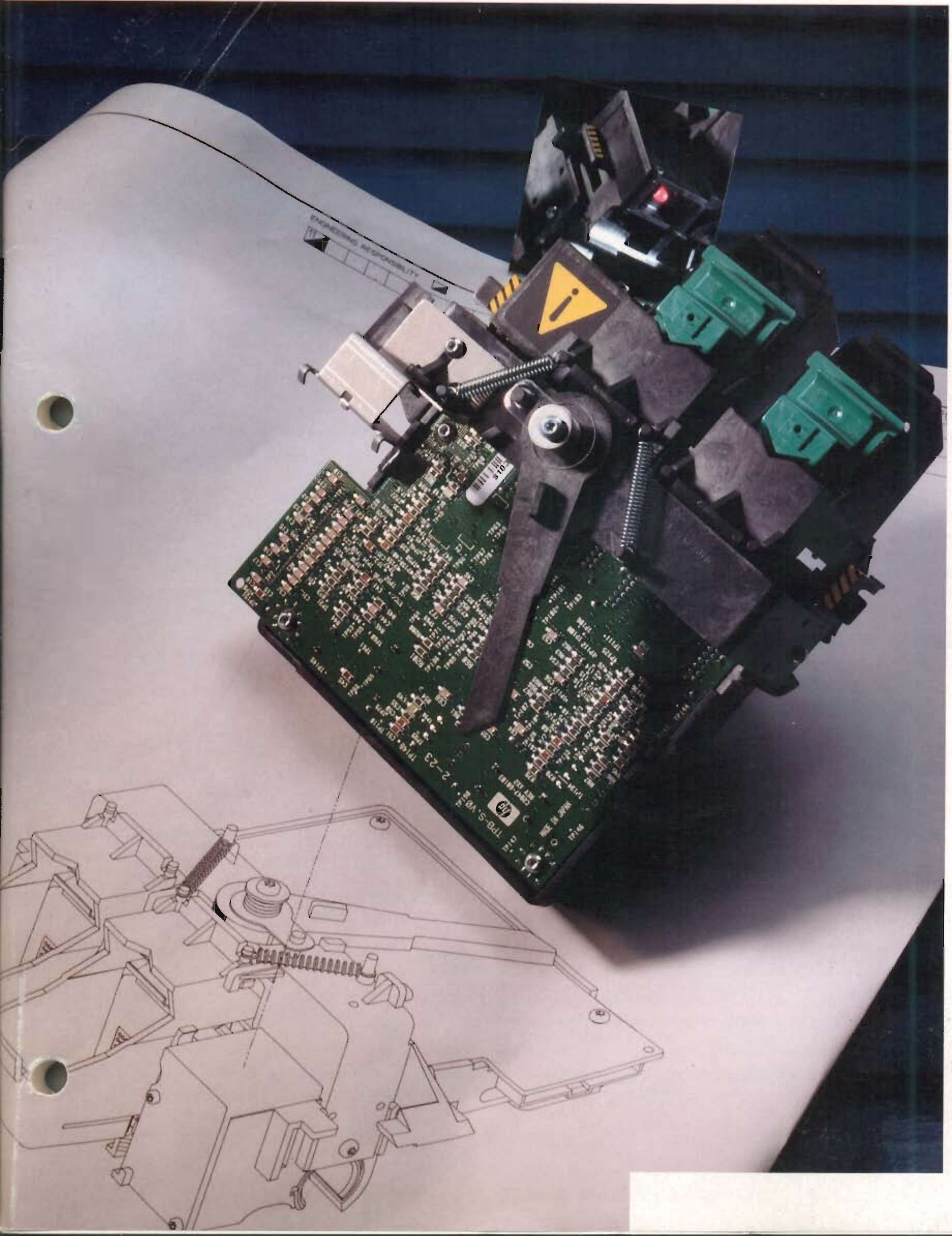
FILE COPY
DO NOT REMOVE

HEWLETT-PACKARD JOURNAL

December 1992



HEWLETT PACKARD

Improved Drawing Reliability for Drafting Plotters

The SurePlot drawing system, a feature of the HP DraftMaster Plus drafting plotter, significantly enhances drawing reliability and unattended plotting ability. The system is based on a noncontact color optical line sensor that verifies the writing of the pens.

by Robert W. Beauchamp, Josep Giralt Adroher, Joan Uroz, and Isidre Rosello

The lines drawn on an HP DraftMaster plotter, HP's high-end large-format pen plotter, compare favorably in precision and smoothness with those drawn by a human. The graphical output appears to be essentially perfect to the unaided eye.[1] The maximum specified pen speed of 110 cm/s and the maximum acceleration of 55.5 m/s^2 have been unsurpassed for over a decade. Although raster printing devices are faster, pen plotters are less costly and are able to satisfy the need for high line quality and multicolor large-format drawings.

To a large extent the value of a drafting plotter is a function of how quickly it can produce drawings of adequate quality, thereby maximizing the daily productivity of the user. Therefore, we saw an opportunity to provide a superior value in pen plotting by reducing the time invested by the user in operating the HP DraftMaster plotter.

HP DraftMaster Plus Plotter

The HP DraftMaster Plus plotter incorporates three types of improvements that enhance the user's productivity:

- Improved reliability, high enough for unattended operation
- Improved media handling
- A friendlier user interface.

These improvements have been made without increasing the cost of the product.

The most common customer complaints about pen plotters have always concerned reliability, mainly pens running out of ink, drying out, or clogging. The SurePlot drawing system is designed to provide a major improvement in the reliability of the drawing system, so that the plotter consistently generates plots without defects, and without requiring constant or frequent monitoring of the pens by the user.

Media handling is improved by roll feed, an automatic media cutter, and a new media tray. Users can retrieve their drawings at their convenience without individually loading every page before plotting or unwinding a takeup spool and cutting plots manually afterwards.

The enhanced user interface offers a redesigned front panel and simplified selection of pens, settings, and drawing quality. Immediate action keys and direct menu access keys give fast access to the functions most commonly used. The information appears in a larger, more easily readable vacuum fluorescent display.

This article focuses on the design of the SurePlot drawing system. The media handling system and the user interface are discussed in the articles on pages 42 and 49, respectively.

Common Pen Problems

Liquid ink pens based on capillary-action ink feed are usually used for final-quality plots and give excellent line quality. However, they require careful handling by the user. Liquid ink pens dry out quickly if they are not capped properly when not in use. Also, commonly used fine-width pens, say 0.25 mm, can experience insufficient ink flow after plotting a certain distance. This is because the relatively high velocity between the pen and the page abrades the paper fibers, which obstruct the end of the pen capillary, ultimately resulting in the clogging of the pen so that it quits writing. This failure mechanism depends on multiple parameters: the thickness of the pen, the type of media used, and the plotting speed and force. Another common occurrence is for one of the pens in the multistall carousel to run out of ink, becoming the cause of an unsatisfactory drawing.

When a pen fails in any of these ways, lines that should have been drawn are missing, and the entire drawing has to be plotted again with a considerable waste of time. The alternative is for the user to monitor the correct writing of the pens, resulting in a significant investment of time.

SurePlot Drawing System

The SurePlot drawing system includes SurePlot pens, a noncontact color optical line sensor, a pen distance monitor, and extra pens. SurePlot drafting pens have ceramic tips for clog-free plotting and regulators to make them leak-free. The optical line sensor allows the plotter to detect the most common pen failures: ink out, dry out, and clogging. The pen distance monitor detects when a pen is in danger of running out of ink on a drawing, based on the lower bound for the life of the pen. Failed pens and pens nearing the end of their lives are automatically replaced.

The system verifies the writing of the pen on the page by periodically sensing lines just placed on the page to verify that the print contrast is adequate for the given pen. The system previously learns the print contrast for a good pen of each type, thickness, and color. If the print contrast is unsatisfactory, the defective pen is replaced and the plot is either restarted from the beginning or retraced from the last good verification, or the plotter halts to allow the user to select the appropriate corrective action.

An improvement of 40 times in the number of drawings not meeting user requirements has been measured for plotters using the SurePlot drawing system over plotters without it. At the 90% confidence level, 998 out of 1000 drawings meet user needs.

Optical Color Line Sensor

The design objectives for the line detection sensor for the HP 759X Series DraftMaster Plus plotters were:

- Low cost. The design should not cost more than the existing digitizer.
- Small size. The line sensor package should fit into a space measuring 1 by 1 by 0.5 inch.
- Low weight. The moving part of the line sensor must weigh less than 4 grams.
- Depth of focus. The line sensor must work over a 3.0-mm range.
- Pen colors. The line sensor must detect black, blue, red, green, violet, aqua, and brown. We wanted yellow and orange but were not willing to pay the added cost to include them.
- Media. The sensor must work with chart paper, vellum, polyester, translucent media, and tracing paper.
- Check with pen. The sensor must focus past the end of the pen tip but not be so low as to touch the platen when the paper feeler is touching the bottom of the pen groove. This objective means that the pen does not have to be stowed while sensing lines.
- Lighting. The plotter must work under normal office lighting conditions (lights off/lights on) or in ambient light up to 1000 lux.

Average User Plot

The performance of a hard-copy device almost always depends on the graphic information being sent to it. This means that an exhaustive performance test of a pen plotter requires that a complete test be run for every drawing in a set of drawings representing all targeted applications. This amount of test effort seemed excessive for testing the SurePlot drawing system for the HP DraftMaster Plus plotters, since we were interested in an average performance measure that would allow us to do summary comparisons with other products.

For this reason, we developed the average user plot, or AUP. This is a single, standard plot that synthesizes the graphic contents of a set of drawings representing the targeted applications at the time it was created (1990). The AUP drastically reduces the test effort and allows easy comparisons between products.

To create an AUP for pen plotters, we chose a number of user drawings representative of state-of-the art drawing complexity and applications. We then extracted the graphic information contained in each of the sample plots. This was done in two ways: by vectorial composition and by external shapes composition. To determine vectorial composition, drawing files were analyzed with a special parser that produces histograms of vector lengths, length plotted, number of pen up/down cycles, and other parameters. To determine external shapes composition, we measured the actual percentages of the total plotting length of each drawing that represented curves, lines, characters, filled areas, and so on. After extracting the graphic information in these two ways, we averaged all the information to obtain the composition of the AUP. We then created a unique drawing that reasonably matches this composition and meets our specific needs.

The cost objective and space limitations proved formidable. No sensors existed that met these aggressive requirements. This made it necessary to develop a custom sensor. The resulting sensor meets all of the design objectives and saves tooling costs by using off-the-shelf parts. Assembly costs were kept low by considering the part assembly from the

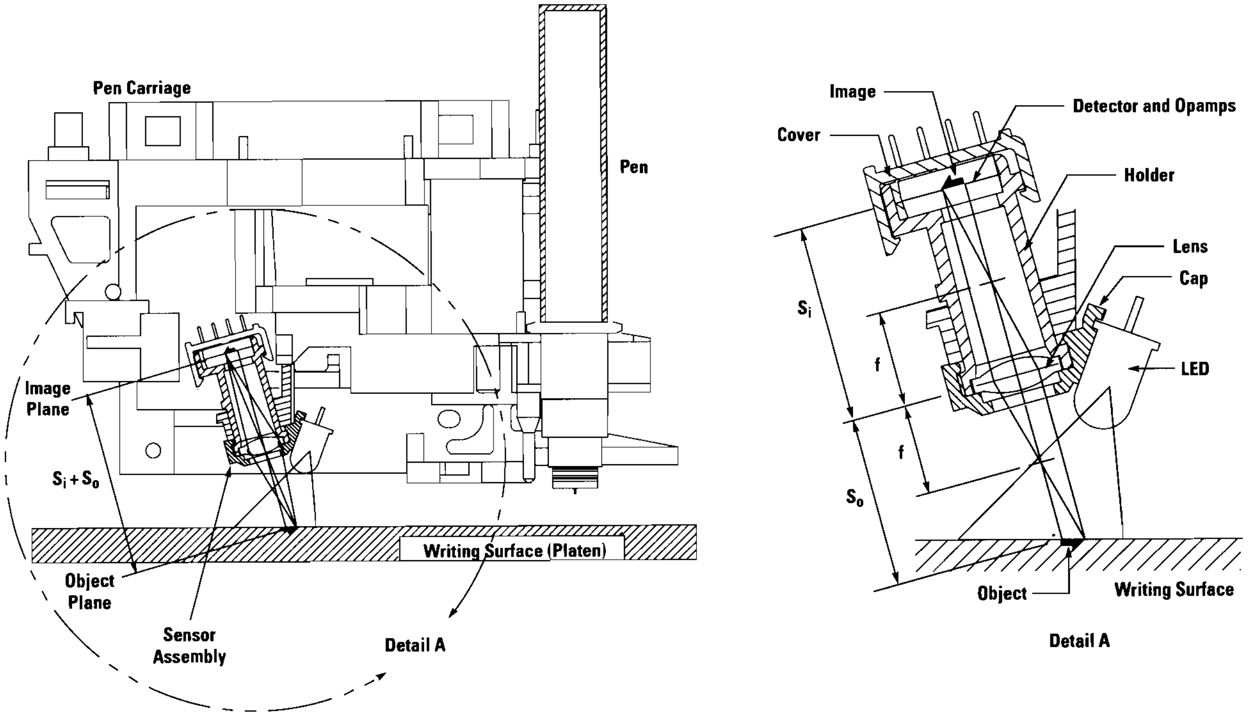


Fig. 1. Noncontact optical line sensor mounted in the HP DraftMaster Plus plotter.

start and talking to the vendors at each step of the design process.

Acceptable Quality Level Index

Acceptance sampling is the set of techniques employed to estimate the quality level of a given set of products by measuring the quality of a limited number of sample products.

Since these techniques are widely applied to incoming inspection of parts to be used in production, most of the literature specifically refers to incoming inspection topics.[1,2,3] Nevertheless, the basic underlying concept is applicable to any product. In our case, we wanted to describe the overall achievable quality of the output obtained during the life of a pen plotter through knowledge of the measured quality of a sample of drawings obtained during a limited time.

Sampling plans can be divided into attribute plans and variable plans. In attribute plans, a sample is taken from the lot and each unit is classified as good or bad. In variable plans, a measurement of a specified quality characteristic is made on each unit.

Since our quality descriptor for every drawing tested was whether it was acceptable, and because we couldn't make any assumption about the underlying distribution of the data, we used the U.S. military standard MIL-STD-105D procedures for sampling by attributes.[2]

The quality index in MIL-STD-105D is the acceptable quality level, or AQL, which is defined as "a nominal value expressed in terms of percent defective or defects per hundred units, whichever is applicable, specified for a given group of defects of a product."

References

1. J.M. Juran, *Quality Control Handbook*, McGraw-Hill Book Company, Inc., 1962.
2. *Sampling Procedures and Tables for Inspection by Attributes*, MIL-STD-105D, U.S. Government Printing Office, 1963.
3. I.W. Burr, *Engineering Statistics and Quality Control*, McGraw-Hill Book Company, Inc., 1953.

Mathematical Model

A simple mathematical model was used in the development of the line sensor. Other scanning sensors focus the emitter light and the detector at the same spot and therefore are very sensitive to changes in height. The basic configuration chosen for the HP DraftMaster Plus line detection system achieves a large depth of field by having the emitter illuminate a wide area and focusing only the detector at a point. This design removes the sensitivity to changes in height.

Fig. 1 shows the sensor design. The LED emits light with an intensity of I_v. The intensity incident upon the media, I_m, is directly proportional to I_v and inversely proportional to the square of the distance between the LED and the media. The light I_m is reflected by the media or ink into the optics, where the lens refocuses the light from the media (object plane) to the detector (image plane). A simple relationship relates the image distance S_i, the object distance S_o, and the focal length f:

$$1/S_i + 1/S_o = 1/f.$$

The depth of focus for a single lens with a focal length f is a maximum at $S_i = S_o = 2f$. This was chosen as the operating point for the HP DraftMaster Plus design.

The variable used to determine whether or not a line is present is the print contrast ratio, or PCR. The PCR is the ratio of the drop in light intensity at the detector resulting from light absorption by the plotted line to the intensity of the general white level of light reflected by the media. Using the PCR to determine the presence of a line removes any dependence on the light intensity.

Ideally, the spot size sensed by the sensor is an infinitesimal point. In reality, the detector needs some area to collect light. The lens also adds to the spot size because it does not perfectly focus the line onto the detector as a result of the aberrations of an inexpensive spherical lens. The spot grows even more when the lens is out of focus because of errors in positioning the object plane.

The voltage output of the detector can be modeled as the convolution of functions representing the line, W, the lens, L, and the detector, d (see Fig. 2). From a system perspective, the input W is a step function of width X_W (the line width). The system is modeled as the convolution of L and d, or $L\star d$, and the output V is the voltage output of the sensor.

The PCR can be calculated using the above model:

$$PCR = (V_W - V_{min})/V_W$$

$$V_W = I_m R_m$$

$$V_{min} = I_m R_i \phi(W) + I_m R_m (1 - \phi(W))$$

$$\phi(W) = \int_{-0.5X_W}^{0.5X_W} (L \star d)dX$$

$$PCR = \phi(W)(R_m - R_i)/R_m,$$

where R_m and R_i are the reflectivities of the media and ink, respectively.

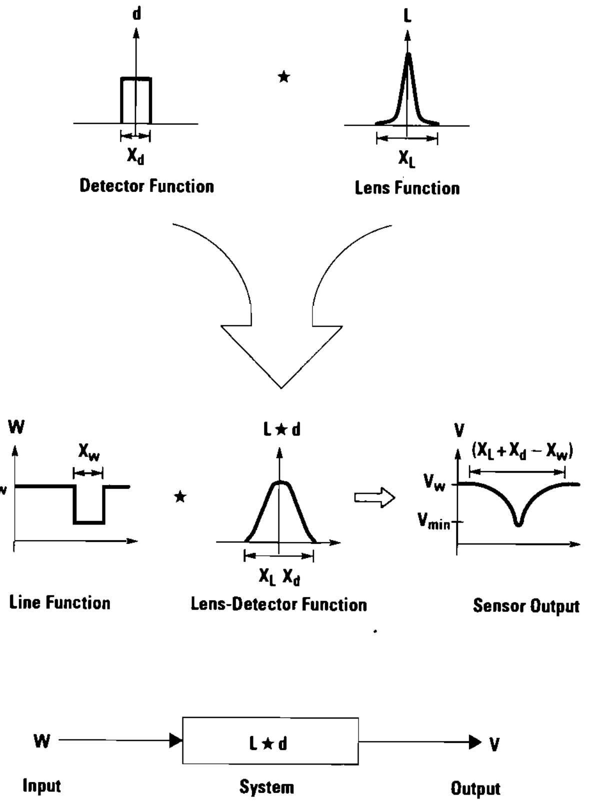


Fig. 2. Mathematical model for the optical line sensor.

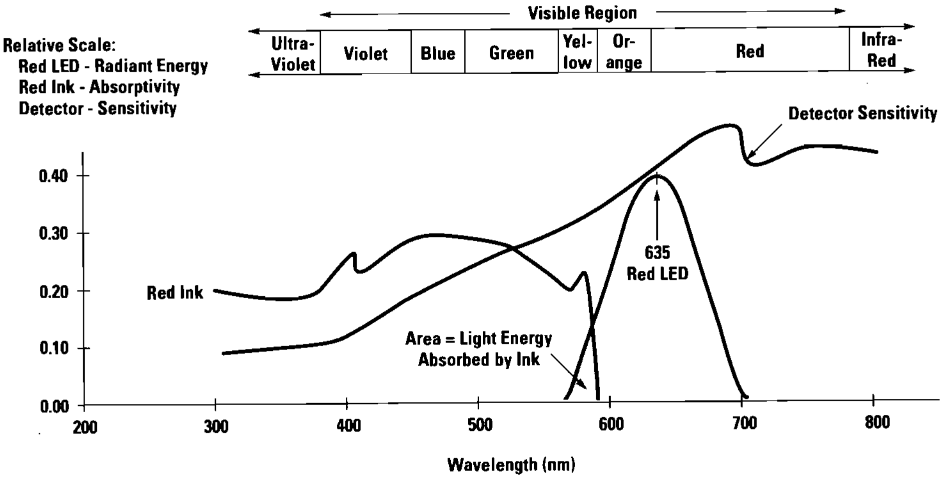


Fig. 3. Red LED and red ink absorbance spectra and detector sensitivity for the noncontact optical line sensor. The shaded area is a measure of the light energy absorbed by the ink.

This analysis demonstrates in equation form how R_i, R_m, and X_W influence the PCR. For example, the absorptivity of the ink is not so important as the difference between R_m and R_i. This implies that some consideration must be given to the fact that media with different reflectivities will produce different PCRs for the same ink. For this reason a white strip lies directly beneath the sensor to boost the PCR for transparent media with very low reflectivities.

As long as the sensor has not been saturated (too much light), the PCR calculation is independent of intensity I_m. This independence was verified by measuring V_W and V_{min} with the room lights on and off.

Light Source Selection for Color Detection

An important consideration, omitted in the above analysis, is the spectral characteristics of the LED, ink, media, and detector. It was tacitly assumed that the reflectivity of the ink, R_i, is the average reflectivity of the ink weighted over the spectral sensitivities of the LED and the detector:

$$R_i = \int_0^\infty R_i(\lambda)I_v(\lambda)I_d(\lambda)d\lambda.$$

Therefore, the PCR can be maximized by minimizing R_i. Because inks are normally characterized by their aborptivity rather than their reflectivity, minimizing R_i implies that the absorptivity A_i needs to be maximized. Thus, a light source should be selected that emits light at wavelengths where the absorptivity of the ink is a maximum. Halogen light sources are very good candidates because of their broad emission spectra but are not used in this application because they get hot and require much more current than LEDs. Blue LEDs are good candidates to detect yellow lines but they are costly, emit very low-intensity light, and would have problems detecting blue lines. Red LEDs have the highest-intensity output but they have a known problem sensing red ink. HP's high-intensity green LEDs were chosen for their low cost, low current, and emission spectrum centered in the visible spectrum. Green LEDs are also capable of sensing green lines because it is difficult to design an ink that absorbs blue light and red light but reflects green light. Fig. 3 shows the spectrum of a red LED compared to the absorption spectrum of red ink. Fig. 4 shows a similar comparison for a green LED/green ink combination. The shaded area represents the value of the above integral and is much larger for the green LED/green ink combination.

Figs. 5 and 6 show the results of sensing different colors of lines with red and green LEDs, respectively. Note the low signal produced by a red LED over a red line, whereas the green LED gives a good signal over a green line.

Detector

The detector has two distinguishing characteristics not found in most other detectors. One is its small size (0.25 mm

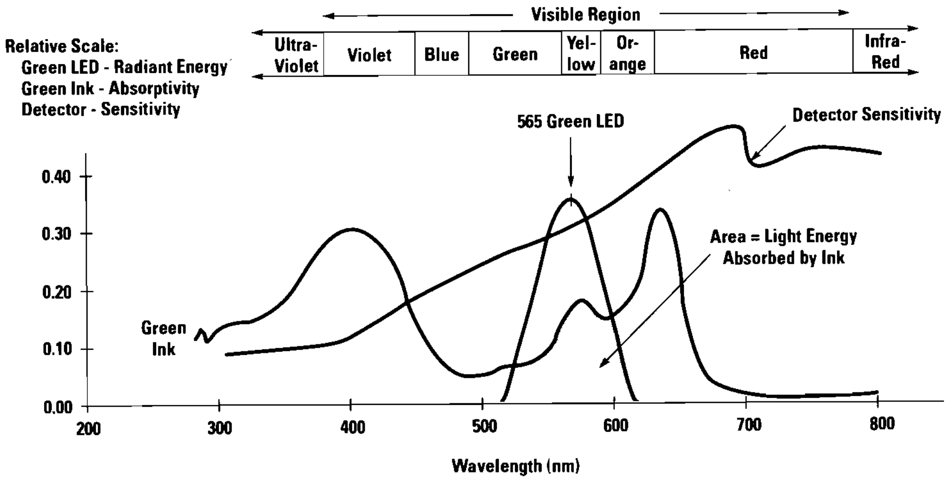


Fig. 4. Green LED and green ink absorbance spectra and detector sensitivity for the noncontact optical line sensor. The shaded area is a measure of the light energy absorbed by the ink.

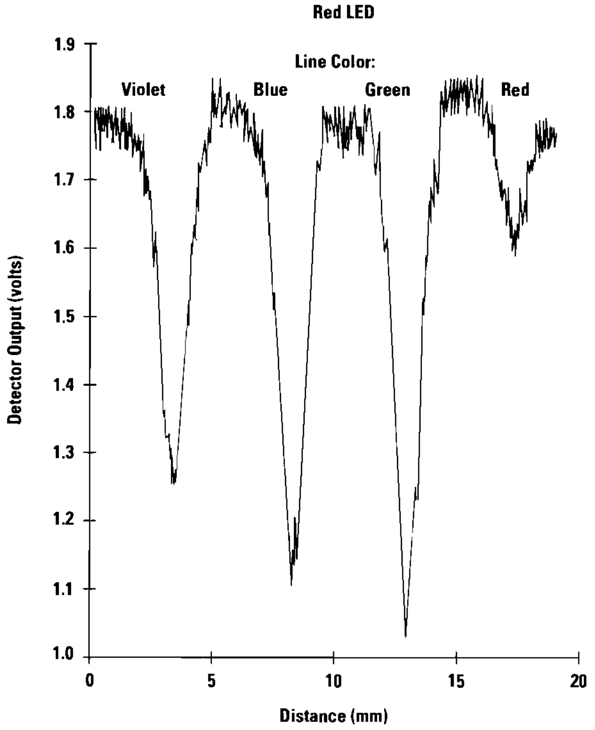


Fig. 5. Detector output for a red LED and violet, blue, green, and red lines.

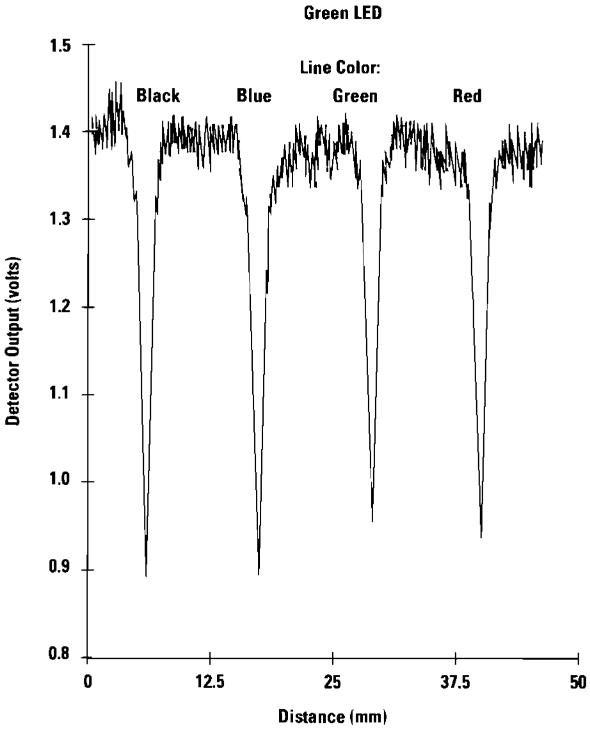


Fig. 6. Detector output for a green LED and black, blue, green, and red lines.

by 0.25 mm), and the other is a two-stage amplifier integrated on the same IC. The small detector size helps keep the spot size small, which is helpful for detecting small lines, while the two-stage amplifier boosts the signal before EMI can obscure it. This is important because the detector signal is sent across a 2-meter unshielded flex cable to the processor printed circuit board. Fig. 7 shows the detector circuit.

Sensor Assembly

Fig. 8 shows the line detection sensor assembly process. All parts are either snapped or press-fit together, thus eliminating messy, unreliable gluing processes and reducing the need for expensive assembly tooling. Excluding the wave solder pass (which is done in batch mode), it takes approximately 20 seconds to assemble the six parts that make up the sensor.

Testing the SurePlot System

Since the degree of attention needed to obtain usable drawings and the quality of the output are highly dependent on the operating conditions, we spent some time before beginning the tests of the SurePlot drawing system attempting to understand or model all of the possible real-life modes of operation and applications that users would be likely to apply.

We split the problem into three aspects: drawing contents, modes of operation, and required output quality. To keep the volume of tests required from becoming unreasonable, we synthesized in a single drawing the average graphic contents of a set of customer-representative drawings. We called the resultant drawing the average user plot, or AUP, (see page 36). Our knowledge of pen plotter customers, accumulated through continuing focus group sessions, customer visits, experience, and other means, allowed us to assume reasonable values for the components of the operating modes: roll feed versus cut sheet, different pen/media combinations, time between plots, workload, applications, and so on.

Finally, we set up the criteria to be able to evaluate the quality of the output obtained and recognize a failure situation. In some scenarios users are expecting high-quality output, so only high-quality plots were counted as acceptable. In other situations, draft-quality plots meet customer needs and are usable, so in these cases draft-quality plots as well as high-quality plots were counted as acceptable. In any

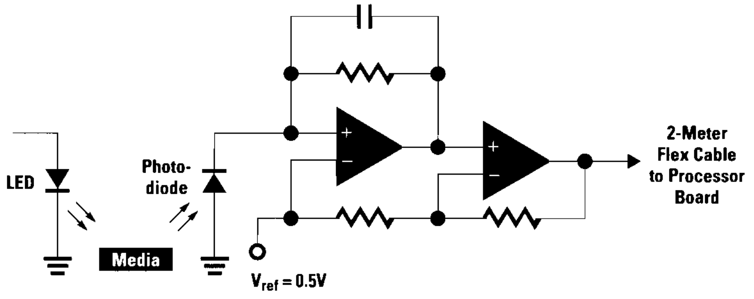


Fig. 7. Electrical circuit of the line detection sensor. The photodiode and the two-stage amplifier are integrated on the same IC.

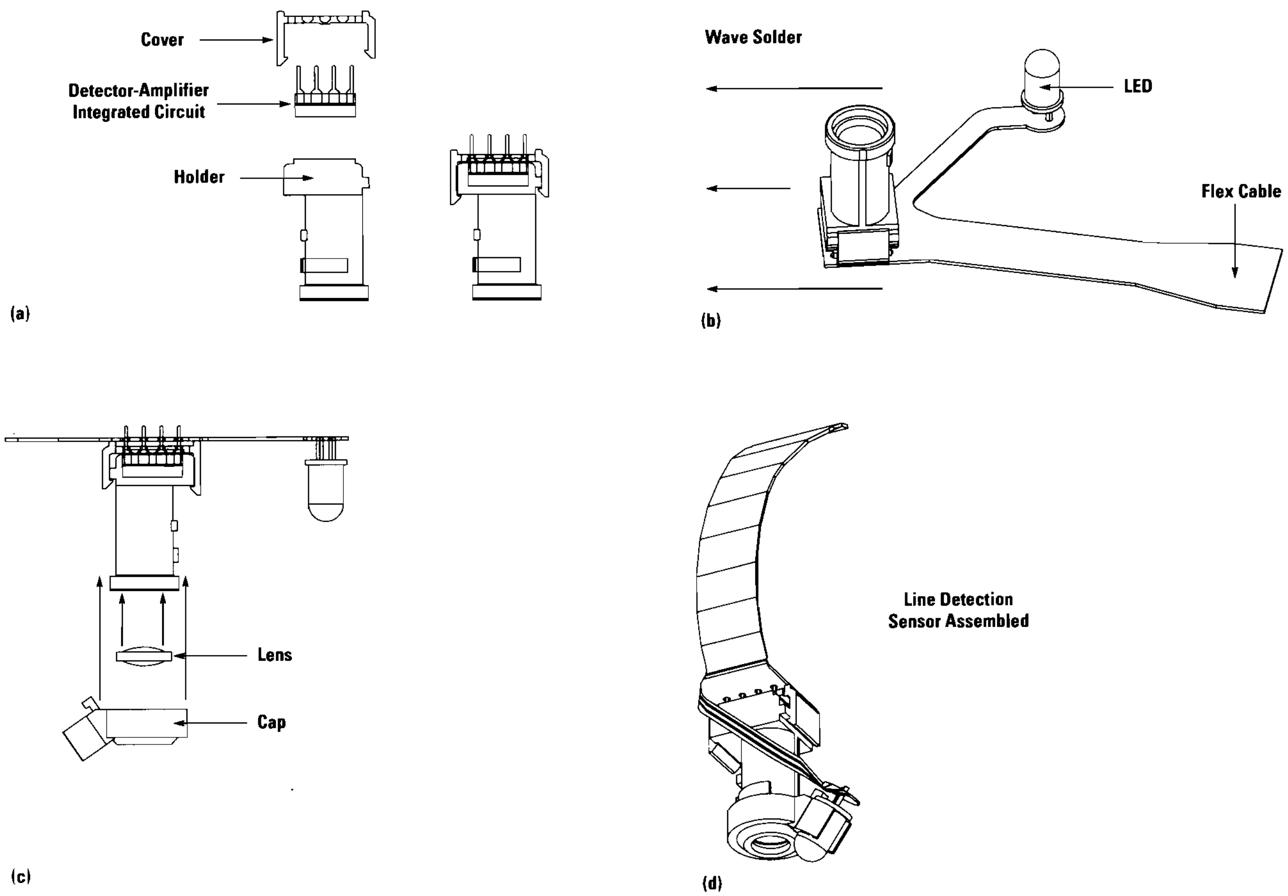


Fig. 8. Line detection sensor assembly process. (a) The detector package is captured between the cover and the holder, which snap-fit together. (b) The LED and the detector IC are wave soldered to the flex cable. (c) The lens is captured between the cap and holder, which press-fit together. (d) The LED is snapped into the cap.

case, a drawing with a missing vector was considered a failure unless it was recovered (only the HP DraftMaster Plus plotter was able to do this).

Test Description

To measure whether and by how much the HP DraftMaster Plus plotter is able to produce more usable plots with less user attention than another pen plotter, we had both an HP DraftMaster RX plotter and an HP DraftMaster RX Plus plotter plotting the AUP eight hours a day during two weeks of accelerated testing in all targeted modes of operation. The tests consumed ten rolls of media and 20 SurePlot and fibertip pens.

The fairly stable and known behavior of the SurePlot pens allowed us to accelerate the testing. These pens were used in a separate plotter until most of their ink capacity had been spent, and were put into the HP DraftMaster Plus plotter near the end of their life. These pens produced all of the HP DraftMaster Plus failure situations recorded. This technique permitted us to extrapolate the results of the 270 drawings plotted to be equivalent to 1500 plots, which represent 150 days of work on the basis of 10 plots per day.

During the test, we collected data on the number and duration of the user interventions necessary to keep the plotters working properly in the selected operating mode, and examined every drawing produced by both machines to classify them as final plots, draft plots, or plots with missing vectors.

Results

Figs. 9 and 10 show results of these tests. The qualitative conclusion is that the SurePlot drawing system allows the HP DraftMaster Plus plotter to deliver a higher percentage of valid plots than its predecessor while requiring only half as much of the operator's time.

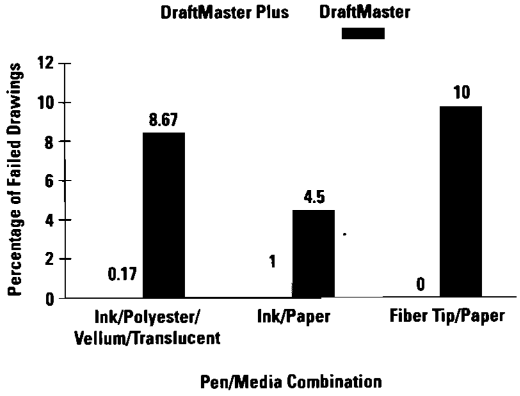


Fig. 9. Drawing reliability test results for the HP DraftMaster plotter and the HP DraftMaster Plus plotter with the SurePlot drawing system.

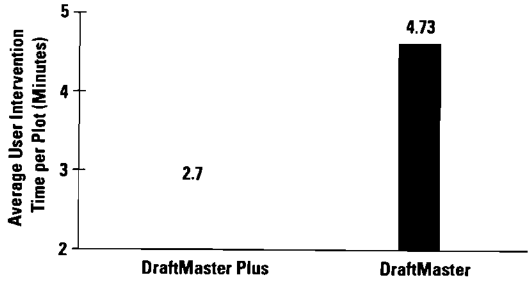


Fig. 10. Average user intervention times measured in the drawing reliability tests.

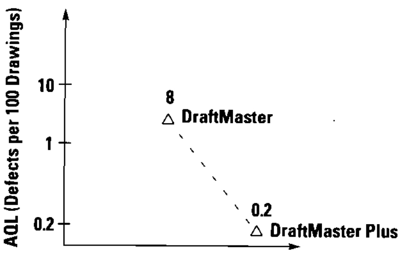


Fig. 11. In the drawing reliability tests, the SurePlot drawing system lowered the AQL (acceptable quality level) by a factor of 40.

Quantitative metrics are highly dependent on the operating mode. From the user's point of view, the question to answer is "How many good drawings can I obtain from a given batch?" To communicate that, we looked for a statistically representative quality index, a single figure that would represent the level of output quality that a customer would experience through the operating life of the plotter. We concluded that the most appropriate index for extrapolating our test data to the life of the plotter was the AQL, or acceptable quality level, defined in a U.S. military standard as "a nominal value expressed in terms of per cent defective specified for a given group of defects of a product" (see page 37).

The AQL results are shown in Fig. 11. The HP DraftMaster Plus plotter had an AQL of 0.2 defects per 100 drawings, compared to an AQL of 8 for the DraftMaster. These results had a confidence level of 90%. This can be interpreted to mean that 90% of HP DraftMaster Plus plotter users, using recommended pens and media, will find that 998 out of 1000 drawings will meet their needs, compared to 92 out of 100 drawings for the DraftMaster. This represents an improvement of 40 times in the percentage of unacceptable drawings over the original DraftMaster plotter.

Summary

The development of a noncontact color optical sensor that verifies the writing of the pens on the actual drawing greatly improves the drawing reliability of a pen plotter. The system operates on an extremely wide range of media, pens, and graphic pattern contents based on its ability to learn the right print contrast. This added functionality does not add cost to the product, since it replaces an existing accuracy calibration sensor. The system can detect errors on the actual traces and recover automatically. On the basis of the AUP and the test results, 90% of HP DraftMaster Plus users will find that 998 out of 1000 drawings will meet their needs. This represents an improvement of 40 times in the percentage of unacceptable drawings over plotters without the new sensor.

Acknowledgments

The authors are grateful to Luis Fernandez, Steve Vanvoorhis, and the entire project team.

Reference

1. M.L. Patterson and G.W. Lynch, "Development of a Large Drafting Plotter," *Hewlett-Packard Journal*, Vol. 32, no. 11, November 1981, pp. 3-7.

An Automatic Media Cutter for a Drafting Plotter

This simple, reliable, low-cost cutter is a classical rotating and linear blade design. It requires no separate drive motors and does not interfere with normal plotting performance. To quantify its performance, cut quality parameters and measurement methods were defined.

by Ventura Caamaño Agrafojo, David Perez, and Josep Abella

One of the main user needs driving the HP DraftMaster Plus plotter development was to reduce the requirement for operator attention to the plotter without increasing the product price. It was very clear that an automatic media cutting device that allowed the user to obtain drawings already cut to the desired size was an essential part of a set of features designed to satisfy this user need. With an automatic cutter, the cost of the product could be substantially lowered for some customers since the existence of a cutting device could eliminate the need to buy a takeup spool.

Determining Customer Needs

To help define the objectives for the cutter development, the technique called quality function deployment (QFD) was used to analyze customer needs and wants. QFD helped to identify the feature set the automatic cutter should have to satisfy customer expectations.

The first step was to collect customer needs by means of focus groups. After analyzing this data, customer needs were classified into groups, each of which represented an overall customer concern such as reliability or performance. The next step was to define the design characteristics that would address the customer needs. To translate cut quality into quantifiable terms we defined a set of measurable parameters. These parameters were used not only to set cut quality goals but also to make comparisons with competing products. The definitions of these cut quality parameters and the procedures for measuring them are presented on page 46. Design characteristics were analyzed based on customer responses and experiments to determine their relationships to the customer needs. On the basis of this analysis, we built the relationship matrix shown in Fig. 1, which indicates how much each design characteristic affects each customer need. Looking at the relationship matrix we could see the features that would meet the customer needs.

One of the most important benefits from using QFD was that it provided the means for concurrent development across all functions—marketing, R&D, quality, and manufacturing. As result of this coordination, design changes were minimal, which helped us to meet our schedule.

Mechanical System Design

Having defined the user needs, the cut quality parameters, the basic design goals, the present state of technology, and the constraints imposed by the product itself, the next step was to determine the cutter design configuration that would be best able to satisfy the various requirements. From an analysis of different possible cutting systems in the light of these requirements, it was apparent that the best solution was a cutter based on the classical rotating and linear blade approach. Instead of being driven by dedicated motors, such a system takes advantage of the plotter's existing drawing driving system.

A breadboard prototype with design parameter adjustment capabilities was built to determine appropriate values of the following design parameters:

- Rotating blade diameter (see Fig. 2).
- Inner rotating blade cone angle (see Fig. 2).
- Cutting speed.
- Rotating blade sharpness.
- Side force. This is the contact force between the rotating blade and the linear blade, which is produced by the compression spring (see Fig. 3).
- Depth of penetration. This is the amount of overlap between the rotating blade and the linear blade (see Fig. 2).
- Shear angle. This is the angle between the rotating blade perimeter and the vertical faces of the linear blade (see Fig. 2).[1]

Using experimental design techniques,[2] we were able with very few iterations to determine that all of these parameters were important, but only three of them were critical, since cut quality was very sensitive to them. These three parameters are depth of penetration, shear angle, and rotating blade inner cone angle. Their acceptable values were measured to be within the following ranges:

- Depth of penetration: >0.1 mm to 0.5 mm
- Shear angle: >0 to 15 minutes
- Rotating blade inner cone angle: >0 to 15 minutes.

With regard to sharpness, the rotating blade edge can be made blunt as long as the contact surfaces between this blade and the linear blade are two sharp points. This is also good for safety and economy.

Although media type is a very important quality parameter, the goal was that all the other parameters should be adjusted to make cut quality as independent of media type as possible. For this reason, no plotter media type ranges were established.

	Design Characteristics												
Customer Requirements	**1**	**2.1**	**2.2**	**2.3**	**2.4**	**2.5**	**2.6**	**2.7**	**3**	**4**	**5**	**6**	**7**
Reliability													
○ Reliability	S												
Performance													
• Cut Quality													
○ No Media Tear			S	S									
○ Clean Cut			S	S				S					
○ Square Edges		W	W	W	S	S		M					
○ Straight Cut		S	M	M									
• Throughput													
○ Quick Cut									S				
• Media Handling													
○ Accurate Size					W	W	S						
○ No Media Waste							W						
• Media Type													
○ Cut All Media Types												S	
Usability													
• Maintenance													
○ Replacement Frequency											S		
○ Easy to Replace												S	
• Human Factors													
○ Quiet													S
○ Safety													S

Design Characteristics:

1. Mean Number of Cuts Before Failure (MCBF)
2. Cut Quality Parameters:
   - 2.1. Straightness
   - 2.2. Waviness
   - 2.3. Perpendicular Deformation
   - 2.4. Parallelism
   - 2.5. Perpendicularity
   - 2.6. Accuracy
   - 2.7. Edge Effect
3. Cut Time
4. Mean Number of Cuts to Replacement (MCTR)
5. Replacement Time
6. Media Types
7. HP Class B2 Environmental Requirements

Relationship:

S: Strong
M: Medium
W: Weak

Fig. 1. Relationship matrix for the DraftMaster Plus plotter automatic media cutting system.

Cutter Operation

The cutter system consists of two basic subsystem devices: the slitting device and the engagement and driving device (see Fig. 3).

The cutter carriage stays in a rest or parking position on the left end of the pen carriage guide arm while the pen carriage is being operated during plotting (see Fig. 4). When the plot has been completed, the paper is moved to the position where the cut is desired. The engagement and driving device, a swiveling lever with a hook on the end, is activated by means of a voice coil, causing the pen carriage to grab the cutter carriage from its parking position and drive it along the pen carriage guide arm. The paper is cut by the slitting mechanism consisting of the rotating blade and the linear blade.

The rotating blade contains a press-fitted O-ring that is held against the media by the leaf springs. Friction between the O-ring and the media causes the rotating blade to rotate while contacting the linear blade at two points (the leading point is the cutting point).

After the cutting operation, the paper is moved out of the cutter path to avoid contact with the rotating blade. The pen carriage moves back to the left end of the guide arm and the cutter carriage is disengaged, leaving it in the parking position. The pen carriage can then be used for a new plot.

The voice coil that activates the engagement and driving device is the same voice coil that moves the drafting pens up and down.

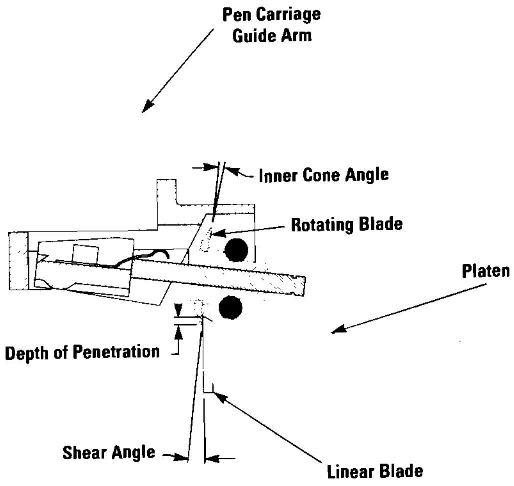


Fig. 2. Design parameter definitions for the cutter, which is a classical rotating and linear blade design.

Slitting Device

As shown in Fig. 3, the slitting device consists of the cutter carriage, the rotating blade assembly, and the linear blade. It uses the same guide arm as the pen carriage, taking advantage of an existing cavity underneath the arm. The cutting speed is fast enough to complete the cut before the cut sheet of media begins its falling motion.

The life of the slitting mechanism depends mainly on the rotating blade because the parts of this blade are subject to the most wear. The blade hub is made of acetal because of its low friction and high wear resistance. Its wear resistance allows it to withstand the high reaction forces against the shaft that are induced by the compression spring. Low friction reduces tangential forces on the media, which could produce buckling and nonstraight cuts.

The geometry of the rotating blade is designed to ensure high cut quality. The side that faces the linear blade is conical so that the contact between the two blades is only at two points. This ensures near-perfect shearing of the media rather than tearing.

For durability, the rotating blade is made of AISI 410 steel tempered to a hardness of 63 Rockwell C. This material also offers high corrosion resistance and reasonable cost. The blade diameter is as large as possible to minimize the number of turns per cut, and the compression spring force is minimized to reduce wear. Since precise sharpening of the blade is not required, no grinding is required in its fabrication, which lowers its cost.

The O-ring is important for obtaining high cut quality because the blade overlap or depth of penetration depends on the diameter of its torus section. It needs to adhere to the surfaces on which it rolls, while avoiding tangential forces on the media. It is made of a rubber that meets these objectives and is resistant to wear caused by the media and media dust.

The O-ring is pressed against the media by the spring-loaded shaft assembly. The friction resulting from this force makes the rotating blade rotate while the cutter translates.

Engagement and Driving Mechanism

Since the DraftMaster plotters can draw along virtually the entire path of the pen carriage, it was necessary to develop an engagement mechanism that would leave the way clear for the plotter to draw. The existing voice-coil mechanism

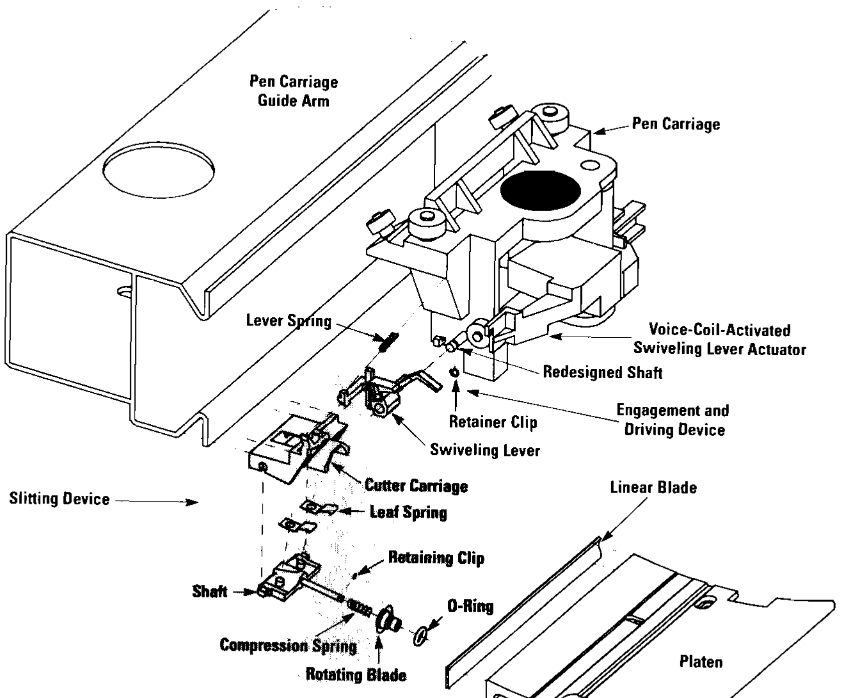


Fig. 3. Exploded view of the automatic cutter, showing the two main parts: the slitting device and the engagement and driving device.

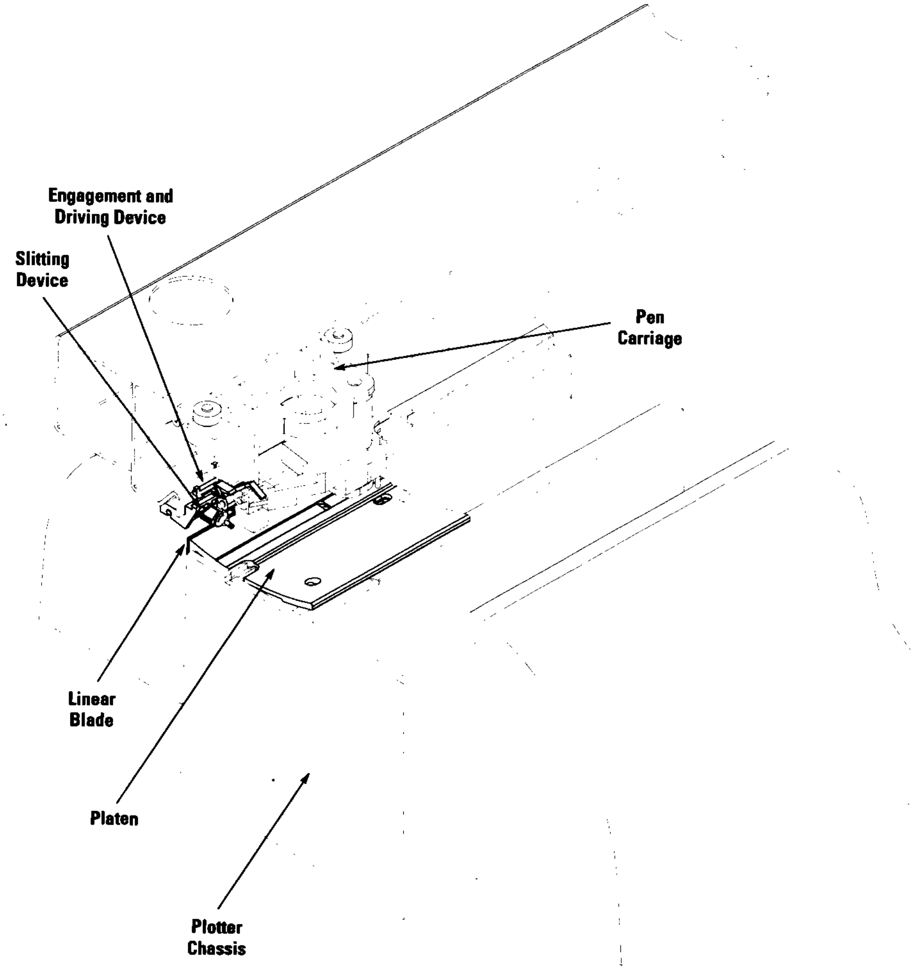


Fig. 4. The parking position of the cutter is at the left end of the pen carriage guide arm.

that moves the pens up and down was easily controllable and turned out to be the best candidate for driving the engagement mechanism. This avoids the need for additional solenoids or motors.

The upper 3 mm of voice-coil travel was not used for any drawing purposes. The HP ME30 three-dimensional mechanical design system was used to find the maximum possible angle of rotation for the rotating engagement lever given just 3 mm of vertical drive motion. Changes in the firmware were made to ensure that the pen carriage does not invade the upper 3 mm of vertical travel during normal plotting.

The pivot axis of the rotating engagement lever is parallel to the horizontal direction of motion of the pen carriage. Thus the inertia forces caused by acceleration of the pen carriage are orthogonal to the lever motion and do not induce any swiveling motion that could disturb the pen carriage and influence drawing quality.

Reliability Testing

The cutter carriage assembly is designed as a consumable part. It has to be replaced when the cut quality becomes unacceptable because of cutter degradation. At the beginning of the project a 5000-cut life was set as a reliability goal for the cutter carriage assembly. This was specified as an MCTR (mean cuts to replacement) greater than 5000 cuts. The MCTR specifies the mean number of cuts a cutter carriage assembly is able to perform before it has to be replaced because of an uncorrectable failure.

According to our user model, a DraftMaster Plus plotter will perform about 67,000 cuts during its 10-year life when used in an environment with a high and continuous workload. Consequently, the reliability goal for the cutter system was set at an MCBF (mean cuts before failure) greater than 67,000 cuts. The MCBF specifies the mean number of cuts the cutter system is able to make before its first failure. The MCBF does not include failures that can be corrected by replacing the cutter carriage assembly.

A nonaccelerated life test was developed to verify the reliability of the cutter system over its lifetime. This test was performed on three pilot-run units with stable parts, avoiding the usual difficulties of prototype testing. The units under test cut HP media continuously at ambient conditions for two months at a rate of 1590 cuts per day per unit. To simulate customer use, three media types were used in the test in the same proportion as an average user: approximately 50% paper, 30% polyester, and 20% vellum. To shorten the test time, strips of media were cut as narrow as possible. Such narrow strips could not fall off on their own, so a set of pressurized air nozzles was installed to blow the strips out of the cutter path.

Definitions and Measurement Procedures for Cut Quality Parameters

Because of the nonexistence of any standard method for measuring the quality of a paper cut, we had to define cut quality parameters, especially those related to the edge finish of the cut, and develop measurement procedures for these parameters before we could determine what the cutter design objectives should be.

According to the literature[1] and our experience, cut quality is usually judged in a rather qualitative fashion, by visually inspecting the fibers that project from the cut paper edge. The less apparent these fibers are, the better the cut quality.

Since the length and density of these fibers are very difficult to quantify, it was necessary to search for other parameters that could represent cut quality as the user sees it and that could also serve our purposes.

It was found that there was a very high correlation between visual cut quality, average edge fiber length, fiber density, and cut waviness. Because cut waviness is reasonably easy to measure, it was selected as one of the main parameters.

Hardcopy media are paper, vellum, and polyester. The cut quality parameters selected define both the quality of the cut edge by itself and the quality of the cut edge integrated in the media. The parameters are:

- Cut edge finish: straightness, cut waviness, perpendicular waviness
- Media geometry: parallelism, perpendicularity, accuracy, edge effect.

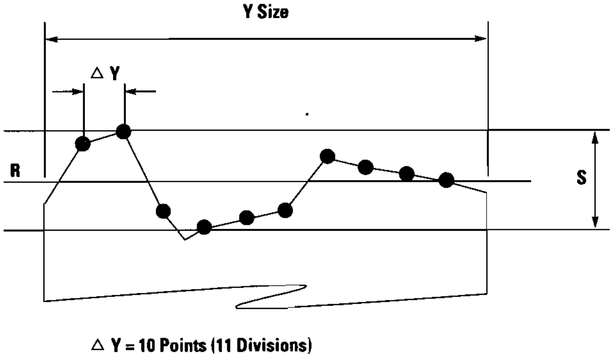


Fig. 1. Definition of straightness S.

Straightness. Straightness (S) is a parameter that measures how straight the edge of the cut is. It is measured as shown in Fig. 1. The location of the edge is measured at ten locations across the cut using a coordinate measuring machine with an optical probe. The regression line R is computed and the maximum positive and negative deviations from R are added to give the straightness S.

Cut Waviness. Waviness (W) is the parameter that measures the undulation of the cut edge along the Y axis. It is measured with a profile projector at 50× magnification in five zones per cut (zone width = 8 mm) as shown in Fig. 2. In each zone, the distance from the deepest valley to the mean crest (where light starts being visible through the fibers) is measured by visual estimation (Fig. 3). Waviness W is the maximum of the five measurements.

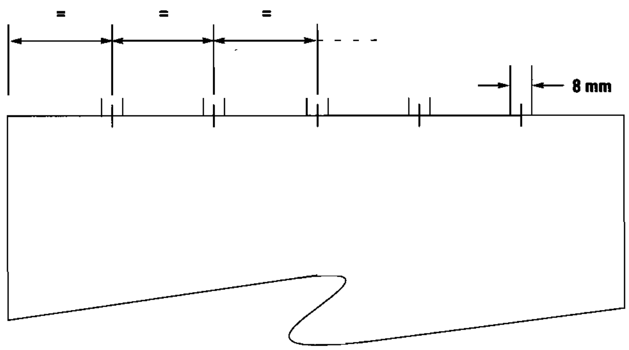


Fig. 2. Measurement points for waviness W and perpendicular waviness Z.

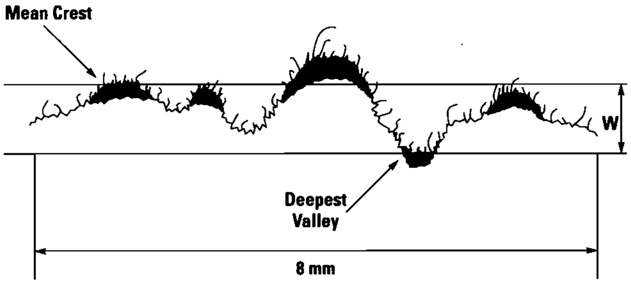


Fig. 3. Definition of waviness W.

Perpendicular Waviness. Perpendicular waviness (Z) is the parameter that measures the undulation of the cut edge along the Z axis (perpendicular to the media plane). It is measured visually using a manual coordinate measuring machine with a suspended knife-edge and a magnifying glass (Fig. 4). Five zones per cut are measured as shown in Fig. 2. For vellum, the characteristic perpendicular waviness is undulatory (Fig. 5); in each zone, the distance between a minimum and an adjacent maximum is measured. For paper, the characteristic perpendicular waviness is not undulatory (Fig. 6) and the total deformation is measured in each zone. No perpendicular waviness is observed for polyester. The perpendicular waviness Z is the maximum of the five measurements.

Parallelism. Parallelism (L) is the parameter that measures how nearly parallel are two consecutive cuts an arbitrary distance apart. It is measured as shown in Fig. 7 using a coordinate measuring machine with an optical probe. First, the regression lines R1 and R2 for the two edges are computed. The parallelism L is the angle between R1 and R2.

Perpendicularity. Perpendicularity (T) is the parameter that measures how perpendicular two consecutive cuts are to a reference line drawn on the paper. A line parallel to the Y axis is used as a reference because of its stability (it depends only on the straightness of the guide arm, while a line at 90 degrees depends on the media tracking, humidity, and other factors). Perpendicularity is measured using a coordinate measuring machine with an optical probe. The regression lines

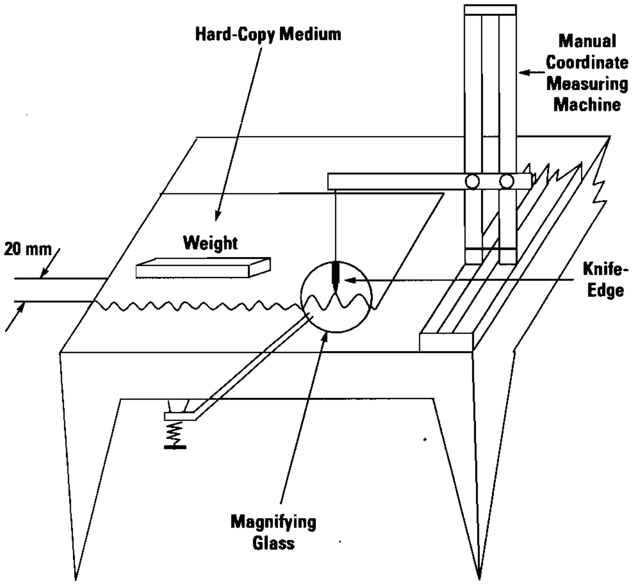


Fig. 4. Measurement setup for perpendicular waviness Z.

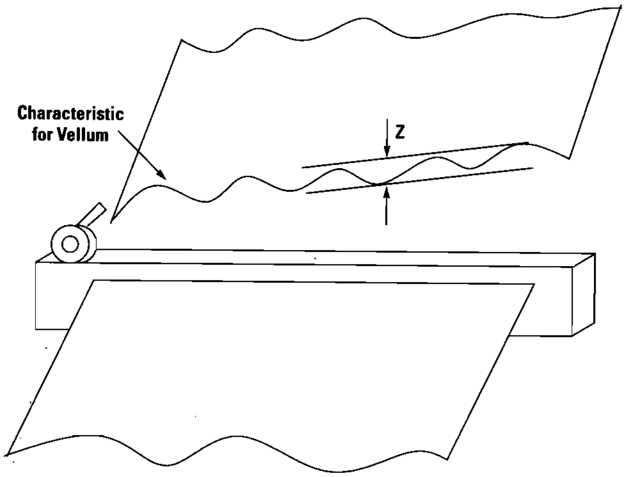


Fig. 5. Definition of perpendicular waviness Z for vellum.

R1 and R2 are obtained as in the parallelism measurement (see Fig. 7). The perpendicularity is the two angles between R1 and the reference line and between R2 and the reference line. Since the reference line is parallel to the Y axis, the ideal perpendicularity is zero degrees.

Accuracy. Accuracy (A) is the parameter that measures the difference between the required and actual sizes of the cut hard-copy media. It is measured using a coordinate measuring machine with an optical probe. One point is defined at each cut edge a distance of 30 mm from the uncut edge of the media at which the cut begins (the offset makes the accuracy measurement independent of edge effect). These points are P1 and P2 in Fig. 7. The accuracy is the difference between the theoretical size and the projected distance on the X axis from P1 to P2.

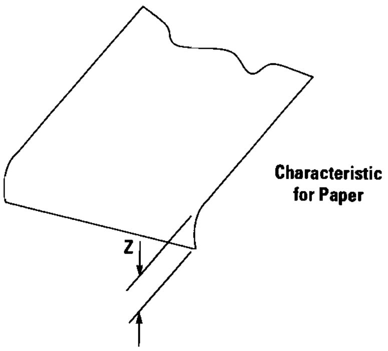


Fig. 6. Definition of perpendicular waviness Z for paper.

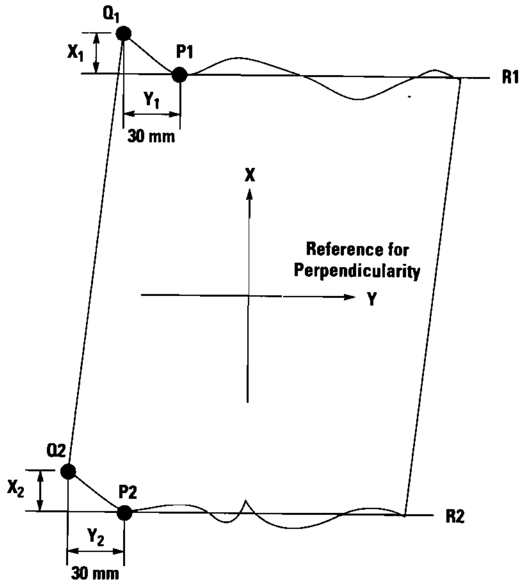


Fig. 7. Definitions for measurements of parallelism L, perpendicularity T, accuracy A, and edge effect Q.

Edge Effect. Edge effect (Q) is the parameter that measures the curve produced at the beginning of the cut. It is measured using a coordinate measuring machine with an optical probe. The starting point of each cut edge is measured. These points are Q1 and Q2 in Fig. 7. The edge effect Q is the maximum of the two projected distances on the X axis between Q1 and P1 and between Q2 and P2.

All measurements are recorded as functions of temperature and relative humidity.

Reference

1. *Wear of Paper Slitting Blades,* Tribology International, December 1980.

Periodically, the cutter carriage assembly, linear blade, pen carriage guide arm, and engagement and driving mechanism were visually inspected for wear. Cut quality degradation was controlled by measuring cut waviness. Cutter carriage assemblies were replaced with new ones when their performance became unacceptable.

At the end of the test each of the three units had accumulated 67,000 cuts. Fourteen kilometers of media had been cut into 200,000 strips. The test did not identify any major problem. Only two risk areas were identified and corrective actions were taken to address them.

Cutter carriage assemblies replaced after 15,000 cuts showed flattened O-rings and some wear on the plastic frame. Cut quality began to degrade when a cutter carriage assembly had performed 10,000 cuts.

On the basis of these results it was concluded that the cutter system meets the MCTR and MCBF specifications.

One of the risk areas we were concerned about was the possible degradation of the coating on the pen carriage guide arm caused by the friction of the cutter carriage. This degradation could result in poor cosmetics and even corrosion under unfavorable environmental conditions.

An accelerated test simulating very intensive use was developed to understand this failure mode. This test consisted of moving the cutter carriage assembly and pen carriage back and forth at high speed along a small span of the pen carriage guide arm without cutting any media. The coating flaked off only in the areas where the cutter carriage assembly and the pen carriage bearing wheels had rolled. No coating degradation appeared in the areas where just one of them had rolled. This test proved that some deterioration of the pen carriage guide arm cosmetics might show up at the half-life of the product when used very intensively. This led to a change of cutter carriage material, which completely eliminated this problem in subsequent tests.

Conclusions

Concurrent development and testing of this system in the development phases, treating it as if it were in production, was key to obtaining a product whose characteristics and performance can satisfy user needs.

This experience taught us that development tools such as QFD are very helpful if used with discretion. It is important to assess the limits within which these tools can be of great value to the definition of the product and beyond which their contribution may be diminished or even become a drag on the development efforts.

The result of this project is a cutting system with the following advantages:

- Independent motors and driving means for the cutter are not required, making it simpler, more reliable, and more economical.
- The cutting mechanism of rotating and linear blades offers simplicity, reliability, and low cost and provides high-quality cuts on different types of media with no need of any mechanism to hold the media. It also provides high durability because it is self-sharpening.
- The cutting system adds little weight to the pen carriage during its normal operation as a drafting device, allowing it to accelerate and stop rapidly and have all of the inherent advantages of a low-inertia design. Thus the same pen carriage can perform both drafting and paper cutting functions while maintaining the original plotter performance.
- The same driving means is used for engaging and disengaging the cutter carriage and for raising and lowering the pens without hampering drawing performance or restricting the paper sizes that can be used.
- The cutting device can be retrofit to plotters manufactured without cutting devices.
- The only maintenance needed consists in replacing the entire slitting device with a new one. This is done when the user observes a degradation in performance or when the plotter warns that it should be replaced (the plotter counts the number of cuts performed). The elastic support system, which keeps the slitting device in equilibrium between the compression and leaf springs, makes it easy (by means of a supplied insertion tool or even directly with the fingers) to separate and raise the rotating blade away from the linear blade to take the cutter out and install a new one.

Acknowledgments

The authors would like to thank Josep Maria Pujol for his contributions to the definition and measurement of the cut quality parameters, Diego Torres, Agusti Comadran, and Joan Uroz for their QFD work, and Felix Ruiz, Joaquim Brugue, Carles Viñas, and Robert Beauchamp for their contributions to the cutter design. Enric Guasch was our consultant on experimental design techniques. Jordi Balderas managed the quality assurance program for the project. Steve Vanvoorhis was project manager.

References

1. *Wear of Paper Slitting Blades*, Tribology International, December 1980.

2. G.E.P. Box, W.G. Hunter, and J.S. Hunter, *Statistics for Experimenters: An Introduction to Design, Data Analysis, and Model Building*, John Wiley and Sons, 1978.

Reengineering of a User Interface for a Drafting Plotter

An existing user interface has been successfully reengineered and plotter usability enhanced by selecting, combining, and adapting software prototype techniques and standard software development methodologies.

by Jordi Gonzalez, Jaume Ayats Ardite, and Carles Castellsague Pique

HP's large-format pen plotter family has been evolving for the last ten years, mainly by introducing new models that enhanced the functionality and performance of previous products. While adding more features and providing better performance, new models have usually required more complex and less intuitive user interaction. To facilitate the use of the new functionality of the HP DraftMaster Plus plotter, it was decided to redesign the plotter's user interface.

The reengineering of the user interface was successfully accomplished by applying and adapting a combination of software development methodologies and prototype techniques. The key steps in the evaluation, design, and development of the new user interface were:

- Development of a user interface software prototype for early evaluation and usability enhancement

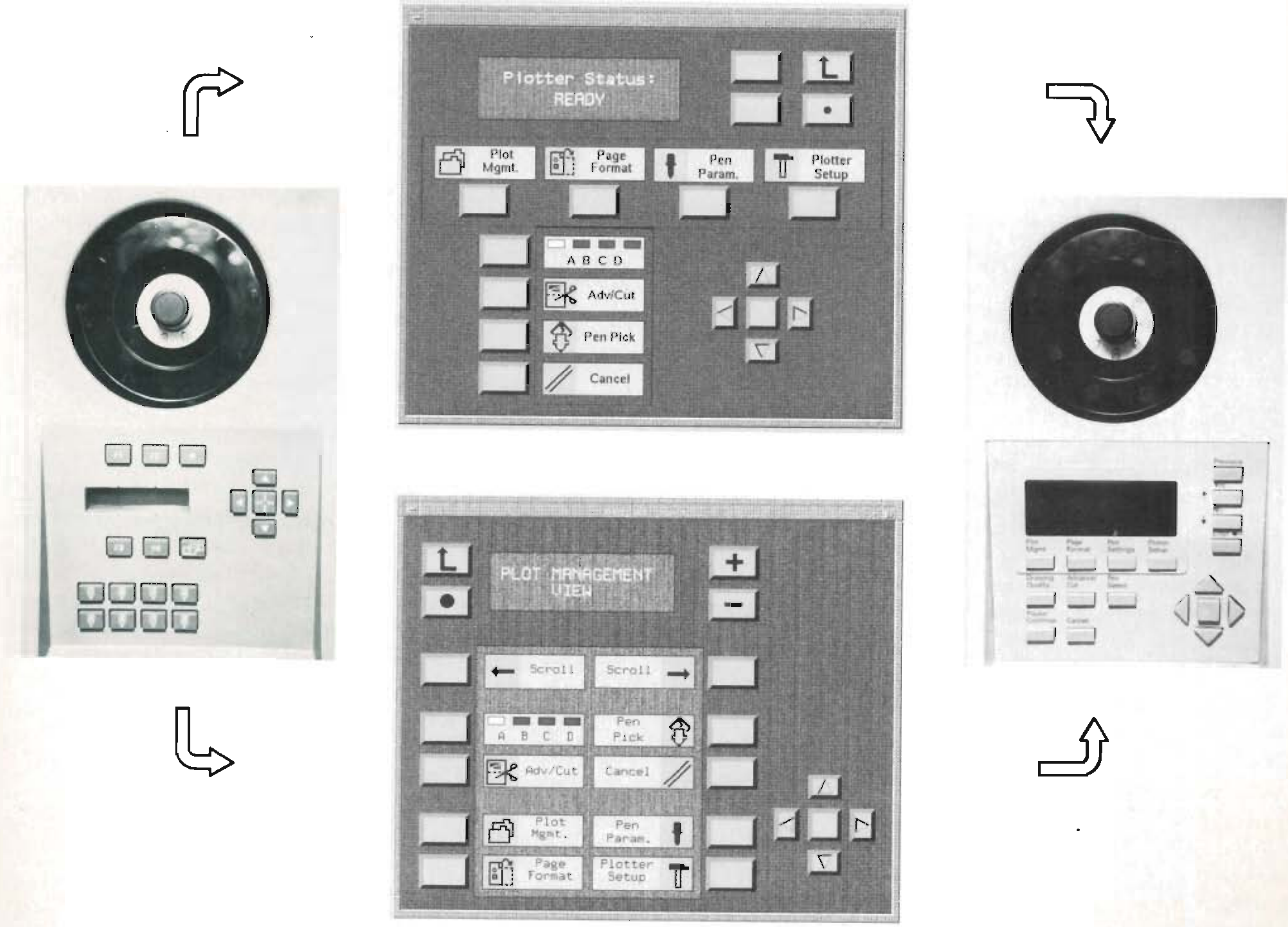


Fig. 1. (left) The old HP DraftMaster plotter front panel. (center) Two of the software prototype options tested. (right) The final HP DraftMaster Plus plotter front panel. In the final design, four keys are reserved for menu navigation and option selection. The most frequently used options are assigned special keys.

- Detailed specification, both graphical and textual, of the user interface menu options, dialogues, and messages
- Design of the user interface for easy localization (translation) of all displayed text
- Use of best practices for design and development: structured analysis, structured design, system testing, design walkthroughs, and code inspections.

User Requirements

Improvement of the HP DraftMaster user interface was identified as a priority as a result of focus groups and customer surveys. The major customer complaint was the readability of the LCD display, especially from different angles. The readability was worst under light conditions that created reflections and shadows. Another area for improvement was the basic user interaction required for menu navigation, function selection, and option setup. Another major limitation was the number of characters allocated to describe each menu option, especially for some languages, such as German and Spanish.

The readability problem was easily addressed by replacing the LCD display with a light-emitting display technology to provide good readability even in poor light conditions. Among the several different display technologies considered, the best combination of cost and features was offered by a vacuum fluorescent display (VFD). To eliminate reflections, the display is covered by dark plastic. The display window has been enlarged to facilitate readability from different angles and greater distances.

To enhance the usability of the front panel, a broader understanding of how users interact with a peripheral was required. Aspects of front-panel design that had to be considered included the layout, type of keys, number of keys, labeling of keys, localization, and aesthetics. Usability aspects included menu selection, option setup, operation feedback, and others.

Our approach to investigating and defining the most appropriate user interface was to use a software prototype tool to build and test different types of front panels.

Rapid Prototyping

The study began with the definition of numerous and highly diverse types of user interfaces. Key people from various disciplines were involved in the definition process: R&D, marketing, quality, product support, and industrial design.

Software prototypes of the different front-panel layouts were generated easily and rapidly using an HP software tool called LogicArchitect. The prototypes allowed people to try different options in a very efficient manner. Very soon the range of options under consideration was narrowed to a small number.

The use of software front-panel prototypes proved to be a very useful and powerful tool with multiple advantages. First, it was useful as a communication tool to describe a particular front-panel alternative. Software prototypes were easily and rapidly sent back and forth between the U.S.A. and Spain through a computer link. Second, as a testing tool, software prototyping allowed the quality department to evaluate concepts early in the investigation phase. This made it possible to design the user interface test suite sooner. The prototype also served as a reference for checking the correctness of the user interface implementation. Third, consensus about which user interface to choose was achieved much sooner with software prototyping, since the advantages and disadvantages of each option were easy to demonstrate. Fourth, the prototype allowed the manual writer to start writing much earlier and helped make the manual more accurate because the writer was able to interact with the prototype. Fifth, the prototype speeded development because the software engineers were able to reuse some of the data structures and texts from the software prototype. Finally, the main advantage was that it was easy to demonstrate the usability of the chosen front-panel option before development.

Fig. 1 shows the old HP DraftMaster front panel, two of the software prototype options tested, and the final front panel.

Risk Assessment

The initial plan was to develop the new user interface in the following pen plotter project. The early consensus on the type of user interface to develop encouraged us to consider advancing its development.

Before committing to the development of the new user interface an estimation of the risk of adding new functionality late in the project was required. This risk assessment was based on estimates of the precision of the current project schedule and the defect removal effort for the project. The conclusion was that the project could be done on schedule, but there would be little margin for error. Therefore, the user interface development had to be done in one cycle with very little time for rework.

The methods used to evaluate the precision of the schedule and the defect removal effort were the keys to a good risk assessment and are explained in more detail in the following sections.

Precision of the Project Schedule. The precision of the project schedule is a measure that can be derived from the project delay over time: the greater the project delay the less accurate the project schedule.

Planned project progress can be measured as the number of planned test cases. A test case is a set of tests done to validate a certain unit of functionality. Actual project progress can be defined as the number of test cases completed up to a given time. The assumption is that a test case is performed as soon as the associated functionality is available for testing. By observing the planned test cases and the actually completed test cases over time one can quantify the project delay and the precision of the project schedule. If a test case finds the functionality under test to be invalid, that test case is considered open until the defects are fixed. Open test cases are not counted as completed. This is because extra time is required to fix the defects, resulting in an extra delay in the project schedule.

The number of planned test cases, the number of test cases actually performed, and the number of completed test cases (actual minus open) can be plotted as functions of time as shown in Fig. 2. Project delay can then be estimated as the difference in time between the date a certain percentage of the test cases are actually completed and the planned completion date. The projection of this difference over time

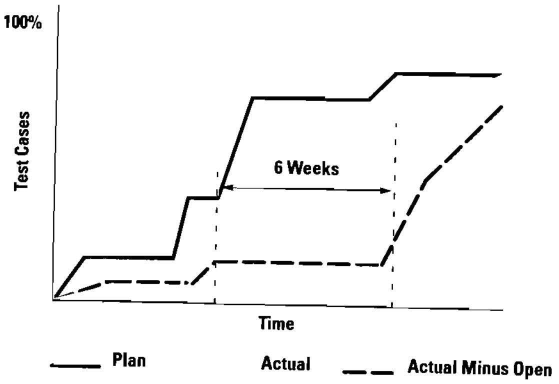


Fig. 2. Plotting the planned test cases (Plan), those actually executed (Actual), and those for which any defects found have been fixed (Actual minus Open) makes it possible to estimate the precision of the project schedule. The estimated delay for this project was six weeks.

for the DraftMaster Plus project resulted in an estimated delay of 6 weeks.

Rework Effort Estimation. The defect removal effort at the point when we were about to undertake the development of the new user interface was estimated as:

Rework effort = (Number of defects) × (Average time to fix a defect).

The average time to fix a defect varies from person to person. It is also related to the laboratory's software development environment. An estimate can be based on the laboratory's defect history by computing the average time it has taken to fix a defect in the past. However, the defect tracking records in the short history of our lab were not sufficient to give an accurate average, so we took the approach of doing our best team estimation. This resulted in an estimate of 2.5 hours per defect, not including test time.

Several statistical techniques are available for forecasting the number of defects. The one that worked best for this project is based on the defect density per test case in each of the system regression tests. The same metric has been used as a project progress measure and as a software stabilization measure. The key benefit of this technique is that it gives a good estimate of the number of defects early in the project.

At the time we assessed the number of defects there were parts of the code that were undergoing the third regression, while others were still in the first. We found that the number of defects per test case and per regression was very close to a straight line, except for the first regression, where the straight-line pattern appeared after 20% of the test cases were executed (Fig. 3). Therefore, we established three conditions for estimating the number of defects:

- At least 20% of the first regression is executed.
- At least 10% of the second regression is executed.
- At least 5 regressions are estimated to be required, based on past project experience.

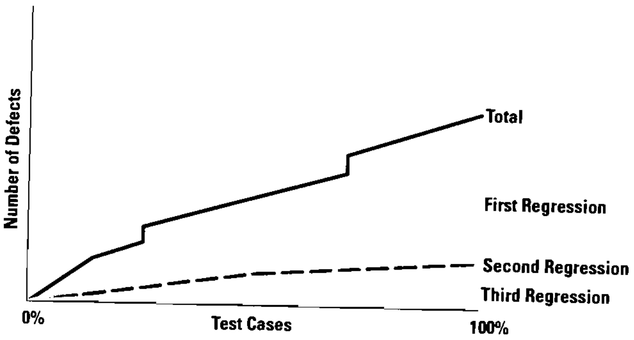


Fig. 3. Number of defects found as a function of test cases executed and the number of regressions. The top curve shows the total defects found for the project.

To evaluate the total number of defects we extrapolated the defects to be found in the first regression at 100% completion, and the same for the second. The the ratio of the numbers of defects in the first and second regressions was calculated. Values for the third, fourth, and fifth regressions could then be determined. The result was that at 20% execution of the first regression, we estimated 96 defects. The actual value at the end of the project was 110 defects for the entire project.

On the basis of the estimated 96 defects and 2.5 hours to fix each defect, the required rework effort was estimated to be 1.5 engineer-months.

Using these techniques we were able also to forecast the number of defects that would be found in the next month. To do this we used our regression test plan and the straight-line relationship between test cases, regressions, and the number of defects. The results are shown in Fig. 4.

Graphical and Textual Specification

As a software good practice, we decided to do as much formal specification as possible of all new software functionality before the design and code development phases. The challenge was to describe the general operation of the user interface formally. To achieve this objective a special graphical syntax, combined with text, was designed as an easy and intuitive way to describe a generic menu-driven user interface. The main objective was to have a working document, the *DraftMaster Plus User Interface Internal Reference Specification* (IRS), which would be easy to review and update for all the different functional areas involved in the user interface development: marketing, quality, R&D, product support, industrial design, and manual writing. The user interface IRS document had to describe the static aspects of the user interface (menu hierarchy, list of options, buttons, etc.) as well as the dynamic and interactive aspects (dialogues, event sequences, option setups).

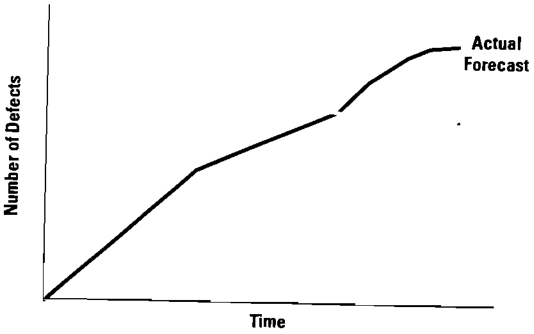


Fig. 4. Number of defects found versus time.

The user interface physical display is one of the basic objects used to describe the user interface. It is represented graphically as a rectangle with shadowed edges. A front-panel button is represented by drawing an outline of its real shape with an icon on top of it for identification. Another element used in the user interface description is the screen, defined as the text being displayed at a particular instant in the user interface display. A screen is described graphically by a rectangular display symbol with the particular text in it. Particular menu options, messages, and option setup screens are graphically represented as screen elements.

The description of a series of user interactions is captured as a dialogue. A dialogue is a sequence of events mainly driven by the user—for example, the setup of an option such as baud rate or the number of copies. To set up a particular menu option, the user goes through a sequence of screens, making selections, setting values, and pressing buttons. Dialogues are described graphically as screens connected by arrows that define a time sequence. In a dialogue, the transition from one screen to another can be triggered by the user's pressing a particular key. This is captured graphically as two screens connected by arrows, with the button graph in between. An example of such a description from the user interface IRS is shown in Fig. 5.

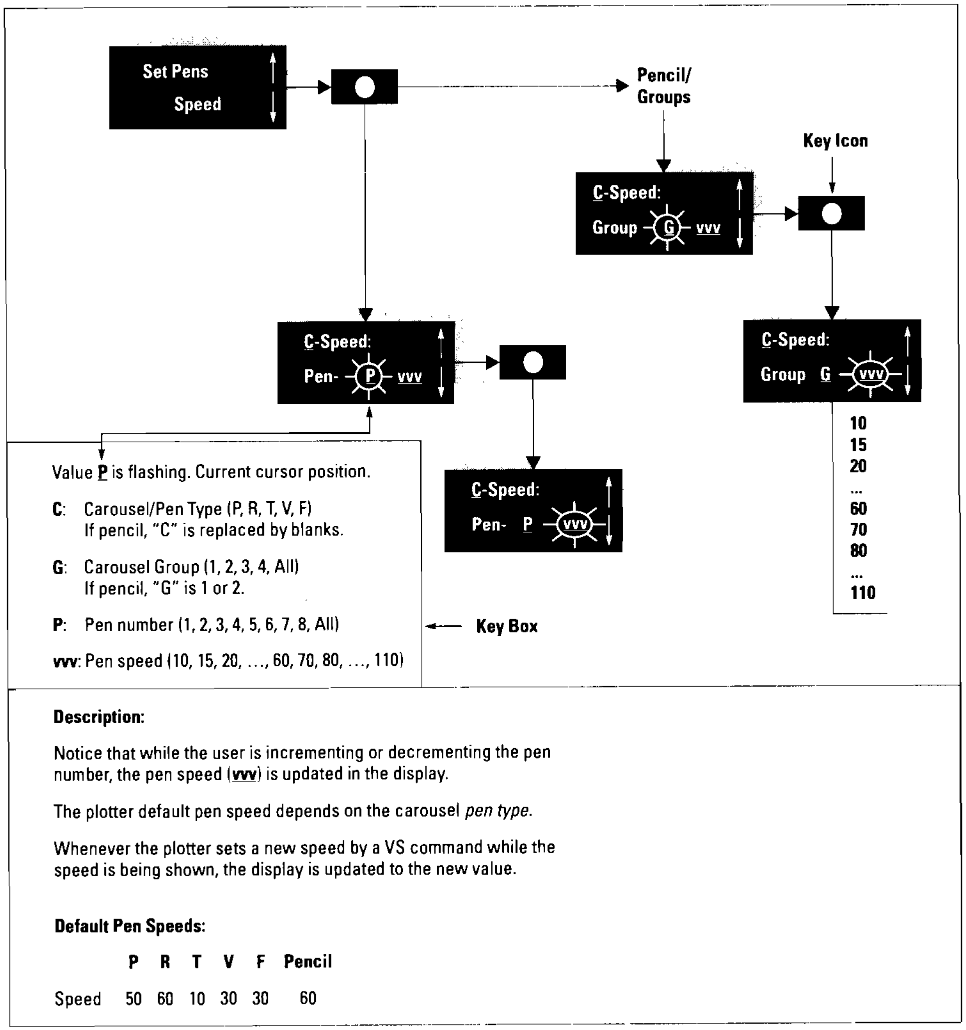
There is a limit on the amount of detail this graphic representation can describe effectively. Details and complementary information are better described as text.

This notation allowed an excellent review of the user interface requirements before the design phase, and facilitated a broad consensus among the different functional areas. The detailed specifications in the user interface IRS also proved to be very useful for system test development. All of the menus and options were grouped into 25 equivalence classes according to the number and kind of buttons to be pressed to reach them, and one menu or option of each class was chosen and fully tested. Status and error messages were tested in the same way, grouping them in classes according to priority, and testing all possible combinations of messages belonging to different classes.

Misunderstanding or incompleteness of an IRS results in software defects. As a result, code and tests have to be reworked. In this project, the detailed specifications minimized this effect and saved a significant amount of time.

Analysis and Design Methodology

Working on a tight schedule, there is little time to redesign previous designs. Things have to be done right the first time to minimize the time to market. With this in mind, we decided to use structured analysis and design practices. The structured analysis, based on data flow diagrams,[1] allowed a team of software engineers to work efficiently. The goal was to minimize the interaction among the engineers while limiting the the analysis to a reasonable level of detail.



Value P is flashing. Current cursor position.

C: Carousel/Pen Type (P, R, T, V, F)
If pencil, "C" is replaced by blanks.

G: Carousel Group (1, 2, 3, 4, All)
If pencil, "G" is 1 or 2.

P: Pen number (1, 2, 3, 4, 5, 6, 7, 8, All)

vvv: Pen speed (10, 15, 20, ..., 60, 70, 80, ..., 110)

Description:

Notice that while the user is incrementing or decrementing the pen number, the pen speed (**vvv**) is updated in the display.

The plotter default pen speed depends on the carousel *pen type.*

Whenever the plotter sets a new speed by a VS command while the speed is being shown, the display is updated to the new value.

Default Pen Speeds:

	P	R	T	V	F	Pencil
Speed	50	60	10	30	30	60

Fig 5. Specification example, with the full graphical and textual description, of the pen speed menu and related dialogues. The combination of graphics, text, and tables proved to be a simple and effective way to describe the dynamic behavior of the user interface menus and dialogues. The display, with the current text message, is represented as a rectangle with shadowed edges. A user event, such as the pressing of a key, is represented by the key icon. Arrows connect the screen sequence to the user action that triggered the event. Special display effects, such as flashing characters or cursor position, are also graphically represented. Lists of values for specific options are represented by variable names with possible values fully detailed in a key box.

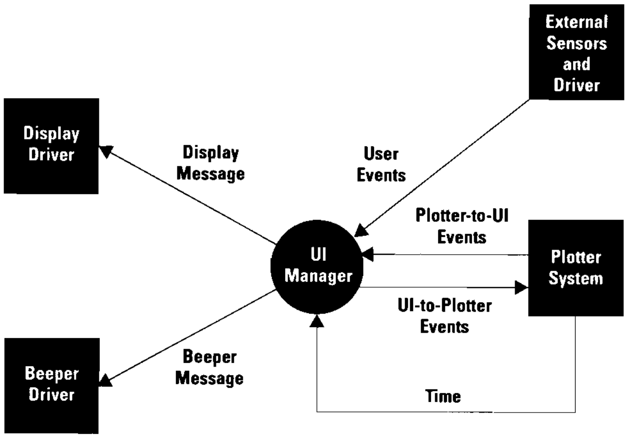


Fig. 6. Context diagram of the user interface manager. It communicates with the plotter system through events. Events are either messages to display, commands to execute, requests for response data, or notifications of plotter state changes. The user interface manager interacts with the outside world through buttons and sensors, a display for menus and messages, and an alarm beeper.

To get a high-level picture of the user interface we drew a context diagram and a data flow diagram. The context diagram (Fig. 6) shows the interaction of the user interface with the plotter system through events and with the outside world through sensors and buttons, the display, and an alarm beeper.

The plotter system is the plotter itself and includes many submodules that can interact with the user interface: I/O, plot management, graphics engine, media handler, vector manager, pen handler, and so on. These modules interact with the user interface manager to notify or warn the user through messages or alarms, show the status of the plotter, show current menu values, set new parameters, and execute user interface commands.

The data flow diagram of the user interface manager (Fig. 7) shows the main procedures in this module: user interface event manager, message handler, menu handler, and display manager. The user interface event manager takes care of all kinds of events, including plotter system, keyboard, and timing.

The menu handler navigates along the menu tree and executes the menus at the tree leaves. It receives the menu events, including keyboard events (a button pressed by the user), data supplied from the plotter, timer events (like an inactivity timeout), and special menus triggered by the plotter from the graphics engine. The menu tree is stored in the menus data structure. The root is the status menu, which shows plotter status, such as ready, busy, paused, paper out, and so on. From the status menu, six main menus are available, four of them directly available from buttons. These have several submenus. At the tree leaves are the menu dialogues—customized menus that can show, toggle, and set parameters or simply trigger with or without confirmation of user interface commands. The user interface allows the nesting over the menu tree of special menus triggered from the graphics engine, such as the digitize menu, and several direct menus for easy access to the cancel, select pen, pause, and other often-used menus. Fig. 1 shows the user interface key layout.

The message handler displays and removes messages. The messages come from the user interface event manager (from the plotter) or are internally generated by the menu handler (to notify or warn the user). The messages data structure holds all the messages, classified by class and priority. Messages of different classes and priorities can be nested. Each class has an associated menu behavior. Informative repetitive messages are displayed until any key is pressed, and are redisplayed after an inactivity timeout. Error messages from the graphics, I/O, and other submodules are displayed until any key is pressed. Informative timeout messages are shown for a few seconds to inform or warn the user. User action request messages require the user to press the **Enter** button to confirm. Progress status messages

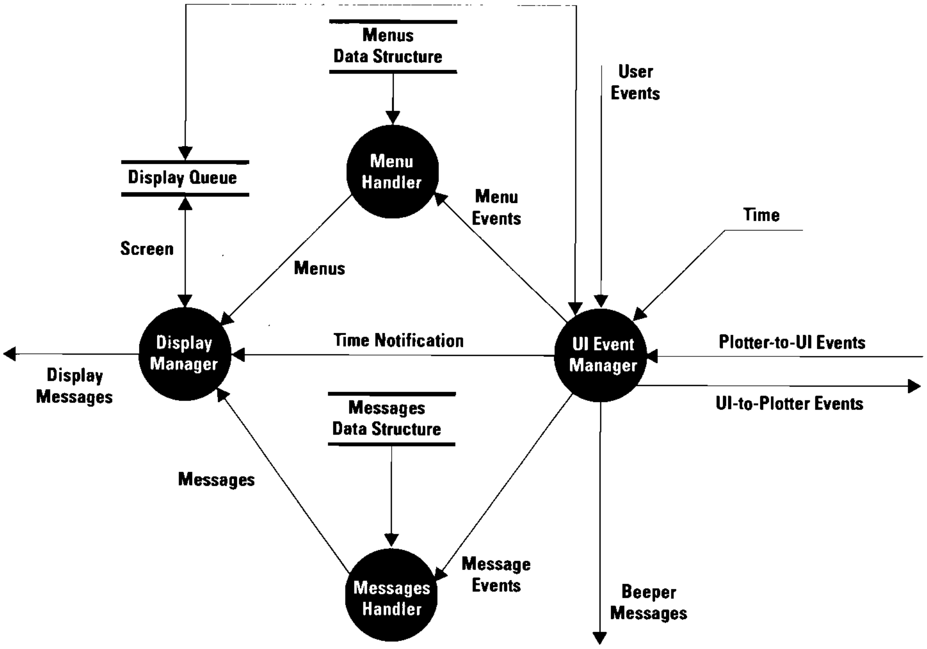


Fig. 7. Data flow diagram of the user interface manager. There are four procedures. The user interface event manager looks for events from the plotter system, the keyboard, or the sensors. It also watches for timing events. The message handler manages messages coming from the plotter or from the user interface on a priority basis. The menu handler navigate along the menu tree and executes the menu dialogues at the menu leaves. The display manager manages the display queue and displays the menus and messages on a priority basis.

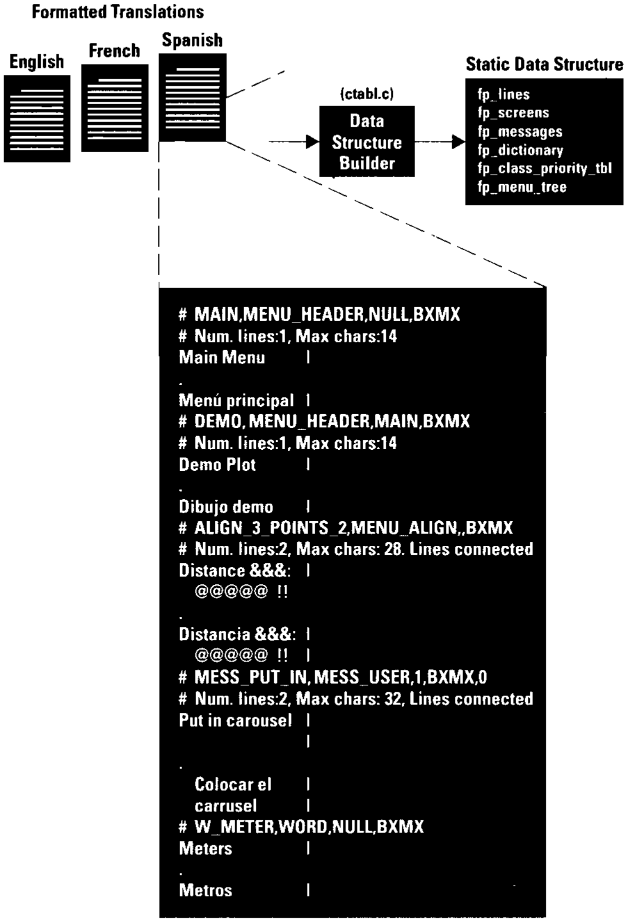
are shown while a critical action is being performed. Critical error messages indicate failures.

The display manager controls the vacuum fluorescent display. It manages the display queue and displays the menu choices. It also sets some of the display options, such as flashing, character set, and brightness.

Data Structures

The next step in the user interface design was the design of the data structures, which are mainly composed of menus, messages, and words. Static data structures were specially designed for easy localization to support the six targeted languages: English, French, German, Spanish, Italian, and Japanese. From experience, we clearly understood the need for an automated procedure to allow further refinement and correction of all the texts and their translations.

Fig. 8 shows the building process for the menus and messages data structures. Fig. 9 shows the text data structures.



```
# MAIN,MENU_HEADER,NULL,BXMX
# Num. lines:1, Max chars:14
Main Menu      I
.
Menú principal I
# DEMO, MENU_HEADER,MAIN,BXMX
# Num. lines:1, Max chars:14
Demo Plot      I
.
Dibujo demo    I
# ALIGN_3_POINTS_2,MENU_ALIGN,,BXMX
# Num. lines:2, Max chars: 28. Lines connected
Distance &&&:  I
  @@@@@ !!
.
Distancia &&&: I
  @@@@@ !!     I
# MESS_PUT_IN, MESS_USER,1,BXMX,0
# Num. lines:2, Max chars: 32, Lines connected
Put in carousel I
               I
.
  Colocar el   I
  carrusel     I
# W_METER,WORD,NULL,BXMX
Meters         I
.
Metros         I
```

Fig. 8. The building process for the menus and messages data structures. The input files are six files with the six localized menu screens, messages, and words. The program ctab filters and merges these files. The output is a C include file with the needed data structures. In the input files, there are menu screens that hold the menu displays used in menu tree navigation, formatted menu screens for the menu dialogues, message screens specifying message classes and priorities, and localized words for the dictionary to be formatted in some menu screens.

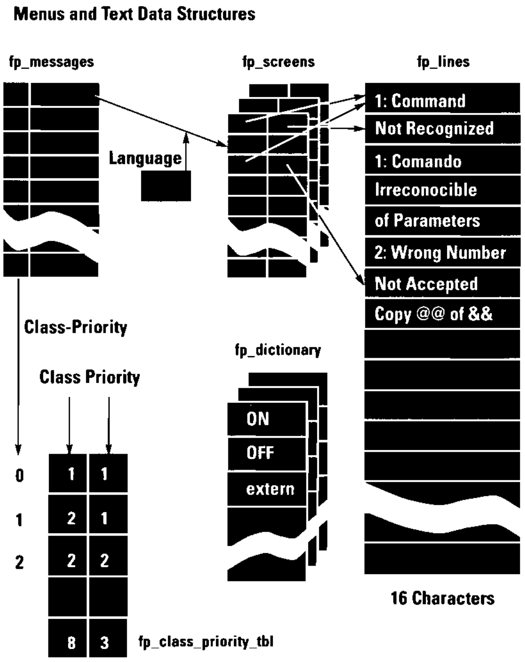


Fig. 9. Text data structures. Messages are indexed by their identifiers. Each entry contains an index to a class priority table to select the appropriate behavior. Another index points to the screen table. For the selected language, there are two indexes to the first and second lines of the message forming a screen. A screen can have parameter fields, specified by the symbols @, &, !, $. The size of a parameter field is specified by repeating the same symbol. When the message is formatted, the parameter values or localized words from the dictionary are substituted for the parameter fields. A similar process is applied to the menus, which are stored in the fp_menu_tree data structure, which has indexes to the fp_screens structure.

In each case, the English version was built first. Some parameters were specified, such as name, type (menu, message, word), menu linkage (for menus), and message class and priority (for messages). There was a field for the English text and another for adding the translation to a single language. This file was sent to the five translators, who had to fill in the translations field.

We created a tool called the data structure builder for automatically building all the data structures from the six files. The tool produces a C include file with all the menu and message structures. When changes had to be made, we just edited the six source files and reran the data structure builder again.

The last step in the design phase was a walkthrough to detect design defects. This walkthrough proved to be very useful. It showed some inconsistencies, but primarily it highlighted a major implementation issue: how to implement the menu dialogues. This issue is covered in the next section.

Implementation

When it came to implementation, there were some inherited constraints that made the reengineering of the user interface more difficult. Most of the code had been written more than ten years earlier when low-level languages were more the rule than the exception. The software system architecture was interrupt-driven and had no operating system for task scheduling and dispatching. The new user interface was constrained to exist in this environment. The plotter system runs as a background process and the user interface is triggered at an interrupt level by a periodic interrupt event. The user interface manager and the plotter system communicate through queues as shown in Fig. 10.

The plotter system was too hard to model, so we decided instead to have a clear, well-specified interface to the user interface queues and then surgically remove the old user interface references and add the new ones.

The user interface manager is scheduled through a periodic interrupt. It is basically a state machine with states, input events, and outputs. When activated, it checks for an event (button press, timing, or plotter) and handles the menus or messages depending on its state and the event.

Another limitation, because of the lack of an operating system, is that the dialogues (menu leaves) were designed as independent tasks. They are called upon entering a dialogue menu and they end when the user operation is completed. They are implemented in the user interface manager at the interrupt level. Therefore, we had to design a small operating system subset just to put the menu dialogues to sleep when waiting for events and wake them up at the next interrupt. We supplied a local stack so the user interface manager would be better isolated from the rest of the system and could more freely build its own display screens.

The last important issue was how to support the design team of three engineers so that they could work efficiently on the same set of modules and within the schedule constraints. We succeeded thanks to a well-established software management control system for supporting parallel development. This system is based on RCS and includes some scripts to better automate the generation of code releases. It also supported our intensive use of code merges. The data structure builder already mentioned made it easy to update the menus and messages.

Results

The HP DraftMaster Plus user interface reengineering project met its planned introduction date and its main project objectives. The number of software defects was very low compared to previous projects in this lab.

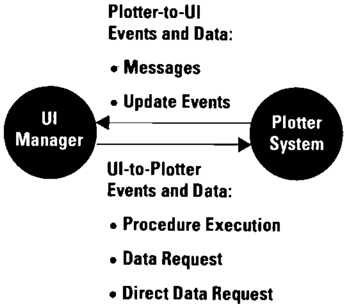


Fig. 10. Interaction between the plotter system and the user interface manager. The user interface can request the plotter system to schedule the execution of a particular function or procedure. The user interface manager can also send a request for data maintained by the plotter system. The plotter system interacts with the user interface manager by means of messages and update events.

The project took 6.5 months. One month was spent in selection of the best proposal. Three months were invested in the definition of the IRS and the system tests—a measure of the amount of specification effort. The final 2.5 months were spent on coding and testing.

The project complexity, measured by the amount of new C code written, was 15 KNCSS (thousands of noncomment source statements).

The number of defects related to the front panel found before introduction was 37. To determine the quality of the specification and coding activity, the defects were classified as specification defects (13%), coding and design defects (84%), or hardware defects (3%). The specification category includes such defects as misunderstandings between team members, side effects of specifications, incomplete specifications, and wrong specifications. The low percentage of these defects is a clear improvement over previous projects, indicating that the specifications were clear enough that every team member was able to understand the expected product behavior.

Defects found in new and old modified code were:

- New code: 38%
- Old code: 62%.

The quantity of code written in the old assembly language was much smaller than the quantity of new code, proving again that the modification of old patched code is much harder than writing brand new code. The prerelease defect density in the new C code was 2.5 defects/KNCSS. This is a significant improvement over previous experience in our lab. At introduction, there were no open defects in the user interface.

Acknowledgments

We would like to recognize the contributions of Juan Jose Gimenez in implementing and supporting rapid software prototypes of the different user interface concepts, Ken Larsen who contributed multiple comments and ideas on the design specification documentation, Herb Sarnoff and Josep Giralt for their significant role in turning the user interface project into a successful reality, and Carles Muntada for his contribution on the user interface project and the help he provided on the revision of this article. In addition, we appreciate the efforts of the engineers at the HP San Diego Technical Graphics Division who helped in the project. Special thanks to the whole project team that made the DraftMaster Plus a real product in the market.

References

1. T. DeMarco, *Structured Analysis and System Specification*, Yourdon Press, 1978.

Bibliography

1. D. Hatley and I. Pirbhai, *Strategies for Real-Time System Specification*, Dorset House Publishing, 1987.
2. M. Page-Jones, *The Practical Guide to Structured Systems Design*, Yourdon Press, 1980.
3. T. DeMarco, *Controlling Software Projects*, Yourdon Press, 1982.
4. M. Ould, *Strategies for Software Engineering: The Management of Risk and Quality*, John Wiley & Sons (UK), 1990.
5. D. Youll, *Making Software Development Visible*, John Wiley Series, 1990.

Authors

December 1992

6 DesignJet Plotter

Robert A. Boeller

Bob Boeller is an R&D section manager at HP's San Diego Technical Graphics Division. A native of Long Island City, New York, he attended the Polytechnic Institute of Brooklyn, graduating with a BSEE degree in 1971 and an MSEE degree in 1972. He joined HP's San Diego Division in 1972 and contributed to the design of the HP 3969A tape recorder. He then served as project manager for the HP 7310A thermal printer and the HP 17623A graphics tablet, and was program manager for the HP 7586A plotter, the DraftMaster large-format pen plotter, and the DesignJet plotter. He is named as an inventor in three patents related to large-format plotters and serves on the board of directors of the Executive Programs of the University of California at San Diego. Bob is married and has two children. He is an avocado farmer and raises show ducks.

Samuel A. Stodder

Development engineer Sam Stodder received his BSME degree from the University of California at Irvine and joined HP's San Diego Division in 1984. He contributed to the design of the paper axis mechanics and worked on print quality issues for the DesignJet plotter. He is named as an inventor in two patents on paper positioning systems. He is married, has a daughter, and enjoys boardsailing, surfing, and sailing.

John F. Meyer

As a manufacturing development engineer, John Meyer was responsible for media development for pen and thermal inkjet plotters, including the DesignJet. Now an R&D engineer at HP's San Diego Technical Graphics Division, he received his BS degree in chemical engineering from the University of Detroit in 1967. Serving in the U.S. Air Force for five years, he did photographic R&D, authored a series of seven reports on photochemistry image quality, and attained the rank of captain. Later, with Snook Corp., he developed photoprocessing equipment. In 1973, he joined the HP Stanford Park Division's Microwave Technology Center and served as a facility, safety, and environmental engineer and as facility engineering manager. John is a native of Roanoke, Virginia. He is married and has one child. His interests include woodworking, photography, and racewalking.

Victor T. Escobedo

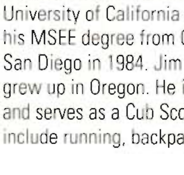
Development Engineer Victor Escobedo joined HP's San Diego Division in 1983. He has contributed to the design of disposable plotter pens, various DraftMaster and DraftPro plotter models, and the DesignJet plotter, including the media stacker. He is named as a coinventor in a patent on a device for stacking cut sheets. A native of Mexicali, Baja California, he received his BS degree in mechanical engineering from California State University at San Diego in 1985. Victor is involved with education, including the Choices program and universities in Mexico. His interests include soccer, diving, and travel.

16 DesignJet Electronics

Alfred Holt Mebane IV

Holt Mebane was the lead firmware designer for the DesignJet plotter. Born in Greensboro, North Carolina, he received his BSEE degree from the Georgia Institute of Technology and joined HP's San Diego Division in 1980. He has designed firmware for the HP 7570A, 7575A, 7576A, and DraftMaster plotters and is named as an inventor in a servo control patent. Holt's interests include golf, amateur radio (N4HR), skiing, and home improvement. He is married and has two children.

James R. Schmedake



A hardware development engineer at HP's San Diego Technical Graphics Division, Jim Schmedake was responsible for the design and development of the processor board for the DesignJet plotter. He joined HP in 1984 with six years of experience in embedded processor design, and has had a variety of assignments including ASIC design and inkjet pen development. He received his BSEE degree from the University of California at Santa Barbara in 1978 and his MSEE degree from California State University at San Diego in 1984. Jim was born in California but grew up in Oregon. He is married, has three children, and serves as a Cub Scout leader. His leisure activities include running, backpacking, and softball.

Iue-Shuenn Chen

Now a principal engineer at HP's Asia Peripherals Division in Singapore, Iue-Shuenn Chen was with HP's San Diego Technical Graphics Division from 1984 until this year. As a manufacturing development engineer for four years, he designed custom production testers and controllers for the HP 7550A, DraftPro, DraftMaster, and ColorPro plotters. Moving to R&D, he was responsible for the DesignJet ASIC architecture and development methodology, the design of the pen interface ASIC and the carriage ASIC emulator, and RFI and ESD testing of the ASICs. He is named as an inventor in a patent on the ASIC architecture. He received his BSEE degree in 1984 from the State University of New York at Buffalo and his MSEE degree in 1987 from Stanford University. Iue-Shuenn was born in Taiwan. He is married and has one child. He is interested in church activities and music.

Anne Park Kadonaga

Anne Kadonaga is a hardware engineer/scientist at HP's San Diego Technical Graphics Division, specializing in ASIC design and computer-aided design methodologies. A graduate of the Massachusetts Institute of Technology, she received her SB degree in electrical engineering in 1982 and her SM degree in electrical engineering in 1984. At MIT she worked as a research assistant and as a teaching assistant in VLSI design. She first joined the HP Data Terminals Division in 1980 as a co-op student, then rejoined the Systems Technology Division in 1984. She designed I/O boards for data terminals and burn-in system hardware for the HP 2700A color graphics workstation. She also worked on a CPU, a cache controller, and a system interface unit for two generations of high-end PA-RISC chipsets for HP 9000 and HP 3000 computers. She designed the processor support ASIC for the DesignJet plotter and the main board for the DesignJet 600 plotter, and worked on print resolution enhancement for the DesignJet 600. She is named as an inventor in several pending patents related to the DesignJet 600. Anne was born in Seoul, Korea and grew up in California's Silicon Valley. She is married and has a son.

24 Pen Alignment

Robert D. Haselby

With HP since 1973, Bob Haselby is a member of the technical staff at HP's San Diego Printer Division. He has contributed to the design of the HP 9872A and 7221A plotters, was electronic project leader for the HP 7580A plotter, and has designed inkjet test systems. For the DesignJet plotter, he designed the automatic pen alignment software and optical systems and the drop detector system. He is the author of a paper on stepper motor control and is named as an inventor in six patents and several pending patents, mostly related to optical alignment systems and DesignJet alignment. His current professional interest is sensor systems. Bob was raised on a farm in Indiana and served in the U.S. Navy submarine service for six years. His BSEE and MSEE degrees are from Purdue University (1972 and 1973). He is married, has two children, and enjoys sailing.

28 DesignJet Chassis

Timothy A. Longust

Mechanical design engineer Tim Longust joined HP's San Diego Division in 1979. He has contributed to the design of the HP 7580A, 7585A, DraftPro, and DesignJet plotters and the PaintJet printer, and is named as a coinventor in patents on the DraftPro ornamental design and the DesignJet chassis assembly. He's now at the Vancouver Division working on DeskJet printers. Born in East St. Louis, Illinois, he received his BSME degree from the University of Illinois in 1979. In 1989 he received an MBA degree from the University of San Diego. Tim is married, has a son, and enjoys hiking, camping, and basketball.

32 Architecture Development

David M. Petersen

For the DesignJet plotter, Dave Petersen served as a mechanical architect and designer and as a manufacturing support engineer. Previously, he was a designer for the HP 7470A and 7475A plotters, mechanical project leader for the DraftPro plotter, and manufacturing engineering supervisor for the DraftPro. He is named as an inventor in two pending patents on the DesignJet chassis and media cutter. With HP's San Diego Division since 1978, Dave recently transferred to the Barcelona Peripherals Operation, where he is a member of the technical staff and a technical contributor. A native of Palo Alto, California, he is married and has two sons.

Chuong Ta

Chuong Ta was project manager for the DesignJet plotter at HP's San Diego Technical Graphics Division. A native of Ha Noi, Vietnam, he holds a BSME degree from the National Technical University of Vietnam, a BSME degree from the University of Minnesota, and an MSME degree from California State University at San Diego. With HP since 1979, he has also been a project leader for the PaintJet printer. Four patents have resulted from his design work. Chuong is married, has two children, and enjoys volleyball, skiing, camping, and hiking.

35 DraftMaster Plus Plotter

Robert W. Beauchamp

Robert Beauchamp is a design engineer at HP's San Diego Technical Graphics Division. He joined HP in San Diego in 1980 and has contributed to the design of the PaintJet and DraftPro plotters and the SurePlot pens and optical line sensor for the DraftMaster Plus plotter. Three patents have resulted from his work on these projects. A graduate of California Polytechnic State University at Pomona, he received a BS degree in mathematics in 1977 and an MS degree in engineering in 1980. Before joining HP he was with the Jet Propulsion Laboratory, where he worked on wideband error correction schemes for Voyager transmissions. He is a member of the ASME and a licensed professional mechanical engineer. Robert was born in Warwick, Virginia. He works with local school children as a Choices volunteer and says that he enjoys running, bicycling, surfing, and "long walks along the beach with my wife and daughter."

Josep Giralt Adroher

R&D engineer Josep Giralt has been with HP's Barcelona Peripherals Operation since 1989. A specialist in computer graphics and firmware, he has contributed to the design of the DraftMaster BX/MX user interface, developed line sensor algorithms for the DraftMaster Plus plotter, and designed and implemented firmware for the DraftMaster Plus. Currently, he is implementing a software development environment and defect tracking system. He holds a degree in electrical engineering from the Polytechnical University of Catalonia and is a member of the ACM and Eurographics. Before joining HP he was with the computer science department of the same university and later with the Siemens software development center. He was born in Barcelona, is married, and enjoys skiing and soccer.

Joan Uroz

As a quality engineer with HP's Barcelona Peripherals Operation, Joan Uroz was responsible for performance testing and competitive analysis in the development of the DraftMaster Plus plotter. Since joining HP in 1989, he has done similar work for earlier DraftMaster plotters. He is a graduate of the Universidad Politecnica de Barcelona and has an Ingeniero Superior de Telecomunicacion degree with specialization in electronics. He is particularly interested in applications of machine vision to print quality evaluation. Before joining HP he developed flexible workcells for Lucas CAV and fluids measurement equipment for Schlumberger. He is named as an inventor in a patent on a remote reading system for home gas meters. Joan is a native of Guadix, Granada. He is married and has two daughters. His interests include skiing, climbing, and other activities in contact with nature.

Isidre Rosello

Isidre Rosello was R&D project manager for the DraftMaster Plus plotter at HP's Barcelona Peripherals Operation. He is named as an inventor in a pending patent on the SurePlot drawing system. Now an R&D program manager, he is particularly interested in process improvement for new-product programs. A native of Barcelona, he received his BS degree in electrical engineering from the Polytechnical University of Catalonia in 1981 and his MS degree in computer science from Stanford University in 1985. Before joining HP in 1988, he worked in new product development for a small company and served as assistant professor of digital systems at the Polytechnical University of Catalonia. He's a member of the ACM. Isidre is married and has a son, and says that in addition to life, friends, and family, he also enjoys reading and bicycling.

42 Media Cutter

Ventura Caamaño Agrafojo

Ventura Caamaño is a manufacturing engineer at HP's Barcelona Peripherals Operation (BPO). He received his BSME degree from the Universidad Metropolitana in Caracas in 1980 and his Master of Engineering degree in mechanical engineering from Stevens Institute of Technology in 1983. He joined HP in 1988, helped build the mechanical infrastructure for the BPO lab, and did feasibility studies of DraftMaster plotter enhancements. He was responsible for the cutter design for the DraftMaster Plus plotter and is named as an inventor in a pending patent on the cutter design. He is an associate member of the ASME. Before joining HP he was involved in the mechanical design of inertial navigation systems with Ceselsa S.A. Ventura was born in Noia, La Coruna, Spain. He is married and enjoys mountaineering, scuba diving, and mountain bicycling.

David Perez

Quality engineer David Perez set the reliability goals and defined the reliability assurance plan for the DraftMaster Plus plotter. A Barcelona native, he joined HP's Barcelona Peripherals Operation in 1989. He has a degree in industrial engineering from the Polytechnical University of Catalunya in Barcelona and is a member of the Collegi Oficial d'Enginyers Industrials de Catalunya. His leisure interests include mountain biking, skiing, scuba diving, tennis, and reading.

Josep Abella

Until recently a metrology engineer in the Barcelona Peripherals Operation quality department, Josep Abella did the metrology engineering for the DraftMaster Plus plotter. Currently a materials engineer specializing in packaging, He joined HP in 1986. He received his degree in industrial engineering in 1985 from Escola Tecnica Superior d'Enginyers Industrials de Terrassa. Before coming to HP he developed fire protection systems with Guardian Iberica, was an industrial engineer with Torredemer, and served a year in the military engineers. He was born in Terrassa, Barcelona, is married, and has a daughter. In addition to spending time with his family, he likes to travel, jog, ski, and play soccer and is a fan of Formula I racing.

49 Plotter User Interface

Jordi Gonzalez

R&D software engineer Jordi Gonzalez joined HP's Barcelona Peripherals Operation in 1988. He has developed HP-GL/2 and pen plotter firmware, most recently for the DraftMaster Plus plotter, and has professional interests in operating systems and distributed processing. He is a member of the IEEE and has a degree in telecommunications engineering (1982) from the Universidad Politécnica de Barcelona. Before coming to HP, he was with Siemens, where he developed computer communication software systems and applications. He is married and is a native of Barcelona.

Jaume Ayats Ardite

Jaume Ayats is a quality engineer with HP's Barcelona Peripherals operation and served in that capacity for the DraftMaster Plus plotter project. He has a degree in industrial engineering from the Polytechnical University of Catalonia (1987) and is interested in quality planning as a tool for project tracking. Before coming to HP in 1989 he worked on automation and process control at the university's Instituto de Cibernetica. Born in Barcelona, he is married and just welcomed his first child.

Carles Castellsague Pique

Carles Castellsague joined HP's Barcelona Peripherals Operation in 1988. An R&D software engineer, he has worked on implementation of the HP-GL/2 graphics language for pen plotters and on the design and development of the DraftMaster Plus user interface. Carles received his BS degree in computer science in 1981 from Barcelona Autonomous University and his MS degree in computer science in 1988 from George Washington University. Before joining HP he was a software developer for the Generalitat Health Department and the World Bank. He's a member of the ACM and is interested in computer graphics. Carles is married and has two daughters. He was born in Granollers, Spain and enjoys mountain biking.

56 Multiprocessor HP-UX

Kyle A. Polychronis

Kyle Polychronis is a project manager at HP's Open Systems Software Division. He joined the Information Networks Division in 1983 as a software engineer, working on the TurboImage profiler and then the HP-UX kernel. He then served as project manager for the HP-UX kernel and the multiprocessor adaptation of HP-UX for release 8.06. He was a consulting project manager for the HP corporate engineering software initiative, and is now project manager for HP-UX open systems I/O. Kyle's professional interests are operating systems, software engineering techniques, and database systems. He received his BS degree in computer science from the University of Utah in 1979 and his MS degree in computer science from the University of California at Berkeley in 1980. Before coming to HP he developed database-oriented UNIX applications for the telecom industry at Bell Laboratories. A native of Salt Lake City, Utah, he is married and enjoys hiking, camping, music, and gourmet food and wine.

Douglas V. Larson

Software development engineer Doug Larson joined HP's Data Systems Division in 1979. He worked on the RTE operating system until 1983, then joined the HP-UX kernel lab, now part of the Open Systems Software Division. For the multiprocessor HP-UX 8.06 operating system, he was responsible for process management and system performance. Born in Minneapolis, Minnesota, he served in the U.S. Navy for six years, attaining the rank of petty officer first class, and attended the University of Minnesota Institute of Technology, receiving a BS degree in computer science in 1978. Before joining HP, he did scientific programming for the University of Minnesota and the U.S. Bureau of Mines. Doug's interests include go (he's a three-dan player), live theater, and ballroom dancing.