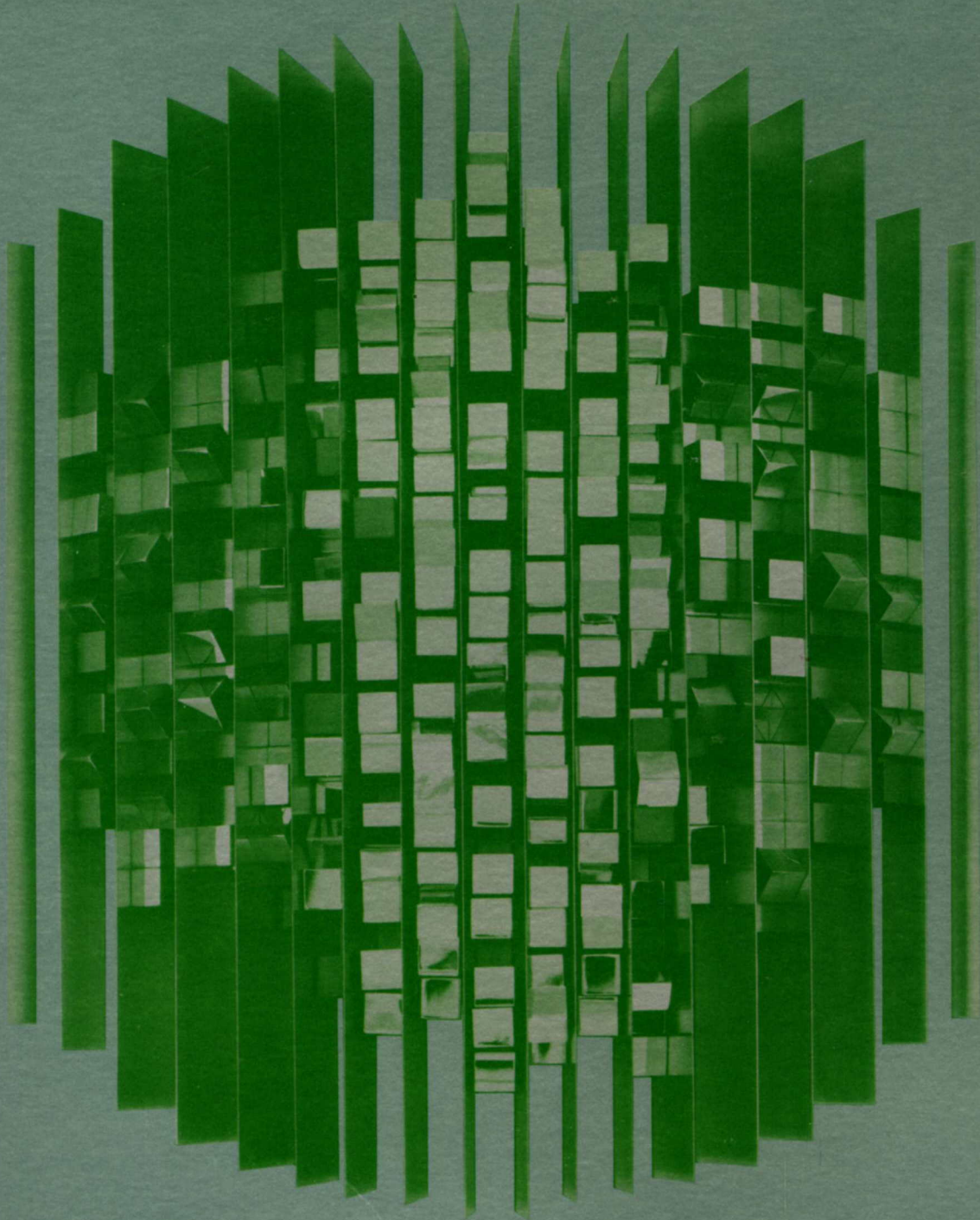


SPERRY UNIVAC
DPS 1100
Display Processing
System



DPS 1100, Designed for Transaction Processing Systems . . .

On-line systems, transaction processing: more and more this combination represents the present—and future—of computer use, as more and more users discover its benefits.

And those benefits are so great that many users overlook the fact that transaction processing, through proper support, can be even more beneficial and economical than it ordinarily is.

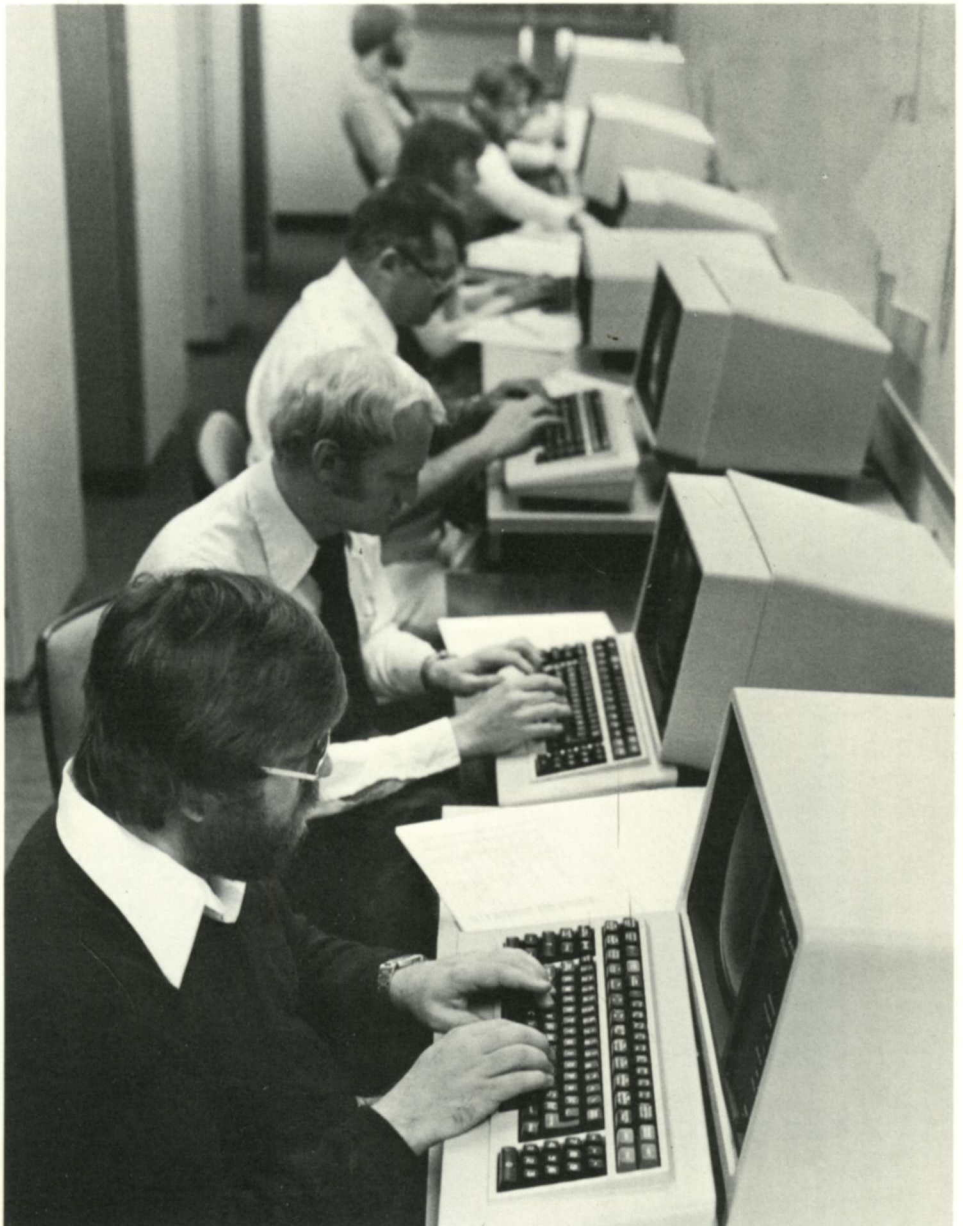
We're not talking about different *means*, we're talking about different *methods*.

The fact is that designing and implementing transaction programs can be a costly and time-consuming process, even for the most experienced programmer. Writing a COBOL source code to define a CRT display (or screen) and its

characteristics not only costs you valuable programming manhours, it also introduces greater possibilities for errors.

The SPERRY UNIVAC Display Processing System, DPS 1100, is designed to attack this problem head-on. It reduces the cost and time of on-line application development by allowing screen formats to be designed, developed, tailored, edited, validated and tested, securely, directly at a CRT terminal, with greater ease than was ever possible before.

DPS 1100 was designed specifically to make your transaction processing system more meaningful and more productive, more quickly.



SPERRY UNIVAC is a trademark of Sperry Corporation.

All specifications subject to change without notice.

The Screen Defined . . .

Usually a CRT terminal is the interactive interface to a transaction processing system. These CRT terminals typically present displays to the terminal operator that give or ask for a variety of information, such as:

- instructions
- blank spaces to be filled in
- prompts
- selections
- headings, field names
- other appropriate data.

In Figure 1, you see a display that is called a "screen." This screen is designed to make the data entry step easier and quicker for a terminal

operator—and also to provide data validation and continued interaction until the complete and correct data has been stored in a record.

There is always a program—or programs—associated with such a screen. This program is responsible for displaying the screen on the CRT, soliciting input from the terminal, checking that input, and entering into a dialogue with the terminal operator as required to complete the session or sequence according to established standards.

It is in the design, development and real-time use of such screens that DPS 1100 can benefit you and your transaction processing system.

Quick, Direct Development . . .

With DPS 1100, screens for transaction systems can be tailored exactly to the needs of your organization and your individual end users. What's more, this tailoring can be performed directly, via a CRT terminal, through the use of the interactive screen definition feature of the DPS 1100.

This powerful feature is complemented by the DPS 1100 screen handler. At the direction of the application program, the screen handler performs the tasks of screen display and input retrieval.

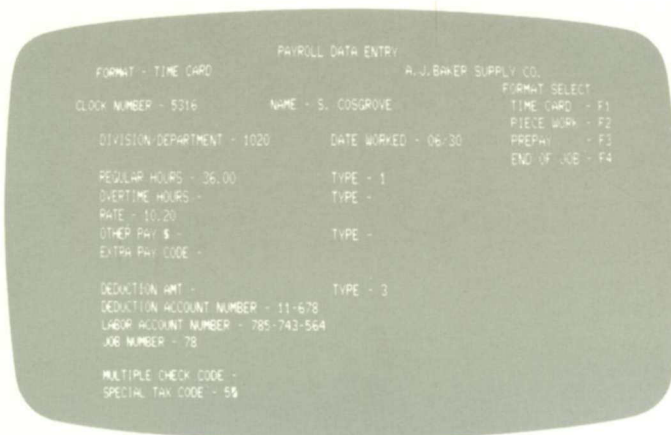


Figure 1: Typical 'Screen' Display on CRT Terminal.

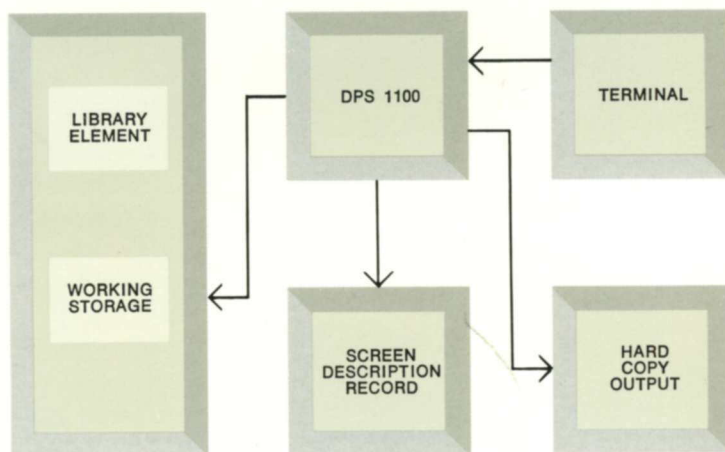


Figure 2: Interactive Screen Generation Via the DPS 1100.



The Advantages of Interactive Screen Generation . . .

The screen handler feature of DPS 1100 also performs critical field editing, validation and security functions according to definitions established at screen-formatting time.

Other features of DPS 1100 include a comprehensive security system, a special test mode for the applications programmer and a multipage output feature for the benefit of the terminal operator.

One way to appreciate the value of the interactive screen generation feature of DPS 1100 is to compare it with the alternative method.

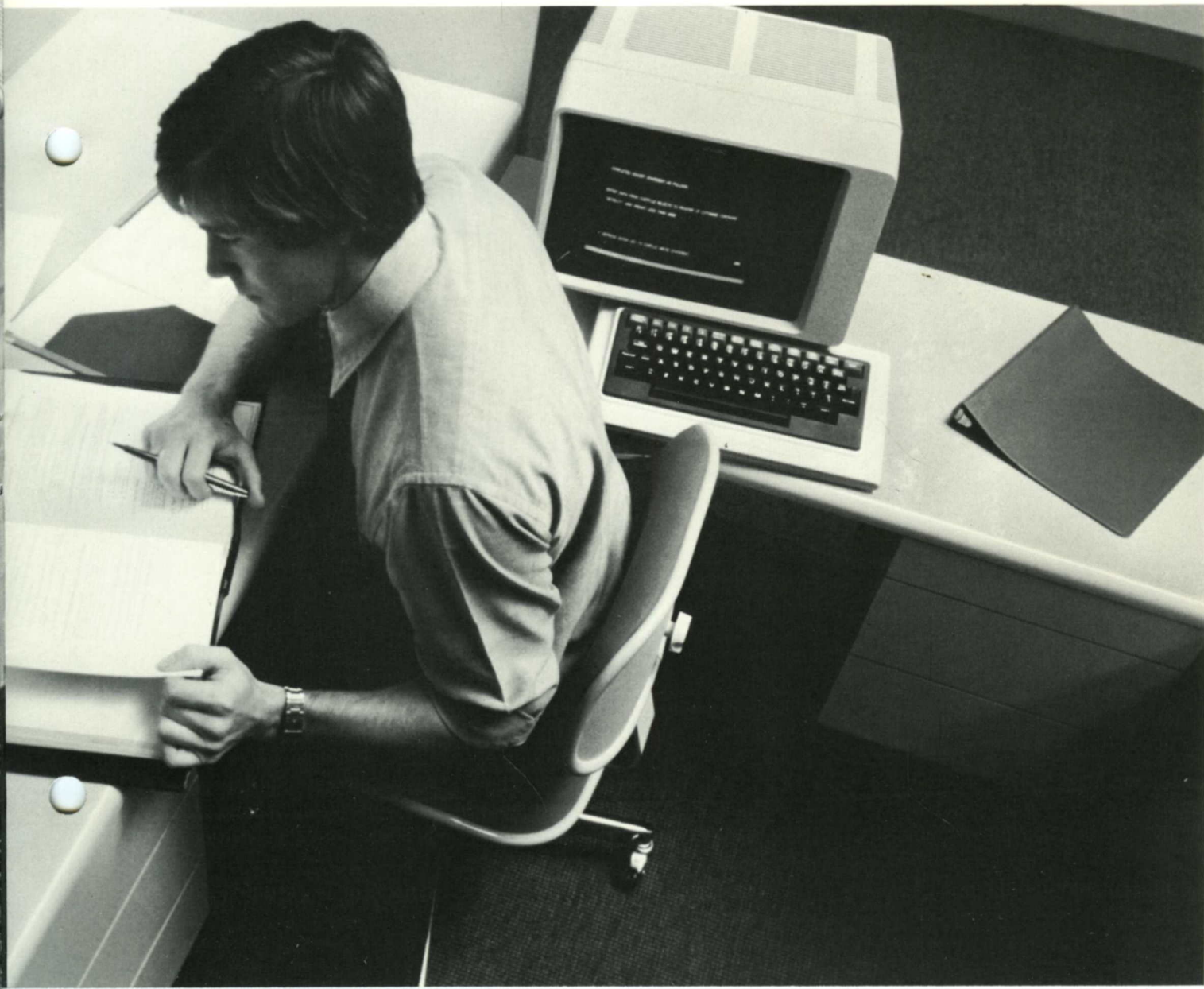
Without DPS 1100, the programmer usually must write COBOL source code to define a display (or screen) and its characteristics as part of his total programming responsibility.

This is a time-consuming process, readily open to the possibility of errors. Thus the time to complete a program is extended, and the maintenance of a program is made more difficult.

As shown in Figure 2, the user of DPS 1100 can format a screen directly at the CRT terminal for which the screen is intended, in a simple interactive session.

Three advantages result from this:

- the program source code is generated automatically, in the form of a library element that can be compiled into one, or several, programs, when convenient
- the screen or display format is maintained in a central file of screen formats
- the entire formatting session is reported by DPS 1100 in a printed record of the screen image, the program source code and other significant information.



Format Development Made Easy ...

With DPS 1100, only three simple steps are required to complete the development of a screen format.

First, you define the screen format by entering the literals and field blanks via the keyboard, exactly as they will appear on the display for the terminal operator. This insures that what the screen developer sees on the display will be exactly what the terminal operator sees during production.

Second, you further identify the fields, which thus far have only been placed in proper position on the display. Type is defined—such as alphabetic, numeric, alphanumeric and others. Other definitions would include such instructions as 'right justified,' 'floating monetary sign,' 'place a comma every three digits,' and such specifications as those concerning mandatory input or output fields.

Third, you complete the definition by adding a final cursor position, screen security level and the choice of an optional conversational mode.

DPS 1100 is Terminal Independent ...

In some transaction processing systems, terminals and applications programs are unfortunately interdependent, because a large share of terminal direction is actually written into the application program. Therefore applications programs must be written to fit terminal characteristics, which can be limiting and inefficient.

With DPS 1100, your application programs become independent of terminal characteristics through the advantages of the separate screen handler.

Figure 3 illustrates the flow of data and control in a SPERRY UNIVAC Series 1100 system using DPS 1100. In an actual system, the screen handler represented by the illustration is an island



of code controlled by the Series 1100 operating system, but accessible through a number of different and concurrent application programs. In use, the application program calls upon DPS 1100 to receive the contents of a record, retrieve an identified screen image from the central screen file, merge the data from the program with the image and finally present the screen image to the terminal operator.

Working in reverse, the DPS 1100 screen handler will cause the input data to be recovered from the terminal and passed back to the application program in a simple, record-oriented form.

The DPS 1100 screen handler can support mixed terminal types, such as the SPERRY UNIVAC UTS 400, the UNISCOPE® 100 and 200 family and the new SPERRY UNIVAC UTS 4000 terminal system.

In addition, DPS 1100 will be able to support new Sperry Univac terminal families as they are announced. The terminal independence feature of the DPS 1100 screen handler should make it easy to change terminals in a system without the need for program revision.

Another advantage of the DPS 1100 screen handler is that application programs can use it in either of the two major interactive processing modes of Series 1100 systems. Series 1100

systems generally use the mode supplied by the TIP 1100 Transaction Interface Package for transaction processing, while program development is generally performed in Demand mode, a timesharing interface. With DPS 1100, programs can be readily developed and tested in Demand mode, interact with the screen handler and some standard screens, and then be used in TIP 1100 mode without modification. This is a great productivity benefit for the programmer.

DPS 1100 also supports a session-control mode, which permits a conversation between the user and a set of transaction programs.

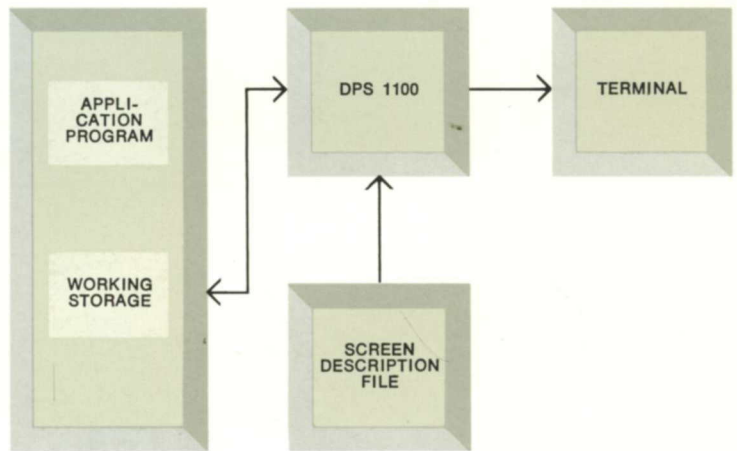
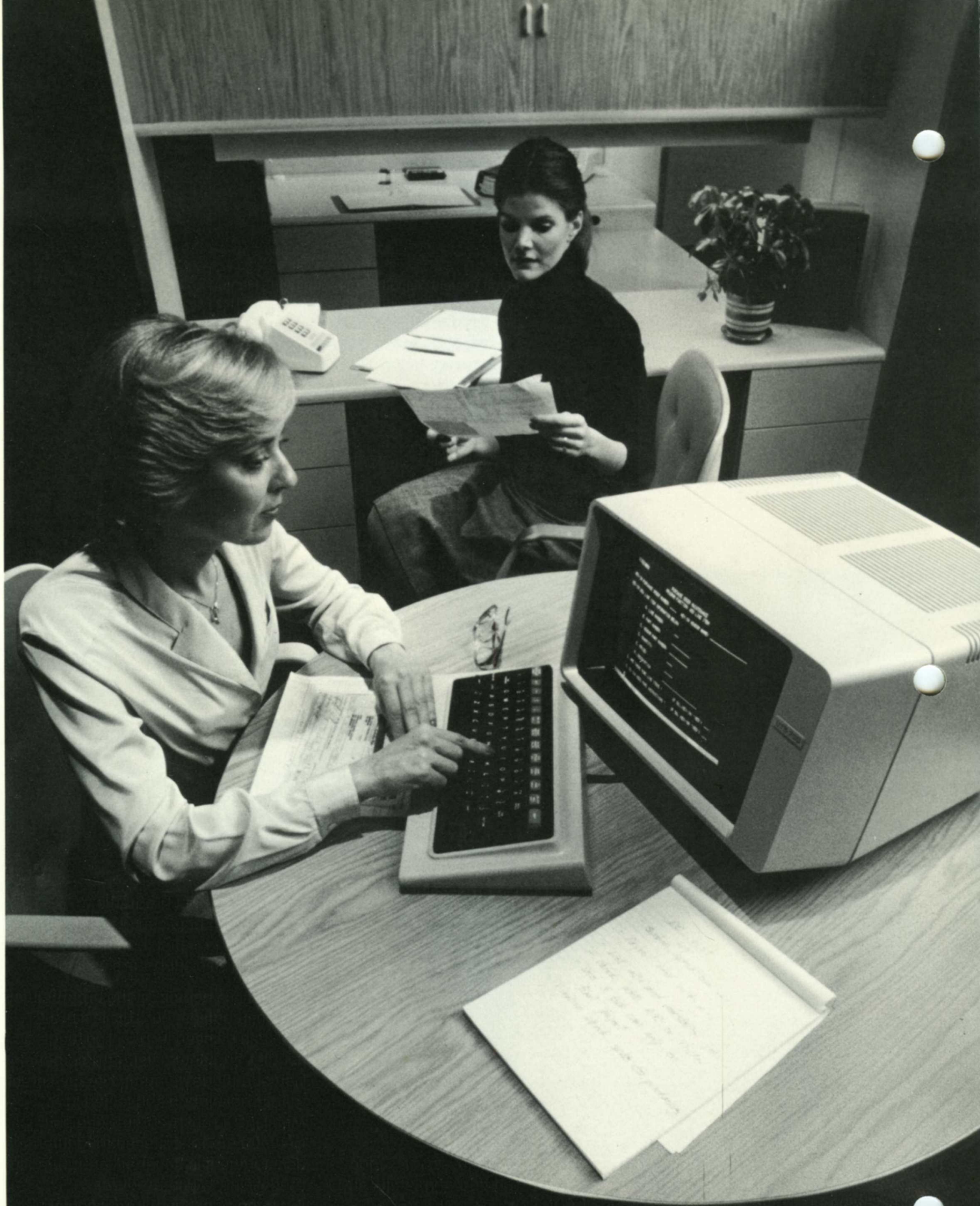


Figure 3: Flow of Data and Control in a Series 1100 System Using DPS 1100 Screen Handler.



Additional DPS 1100 Features . . .

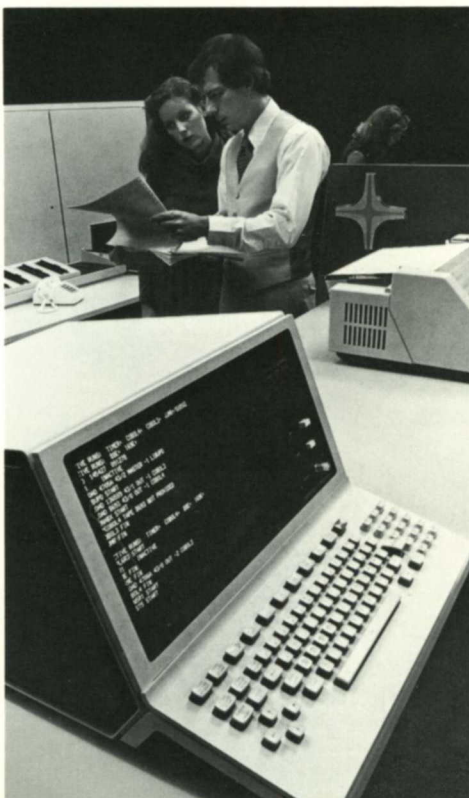
Four additional features make DPS 1100 an extremely attractive software package.

First, screen definition via the DPS 1100 may include attribute descriptions which enable the screen handler to perform editing and validation functions on behalf of the application program. This results in simpler and shorter programs, and it also off-loads work from the program execution.

Second, user profiles may be defined with DPS 1100 for use in a security function. User profiles and associated passwords can be used to control access to and use of screens, and fields within screens.

Third, the DPS 1100 test mode can be used to help the application programmer develop and maintain programs by the simple display of test fields when in test mode, without the need to present that content to the terminal operator in production.

Finally, the multipage capability of DPS 1100 can be optionally used to build a file of screen pages displayed by a program. This may then be accessed by the terminal operator to browse through the output of a given session, for example, by paging backward and forward in the file.



Improved Productivity All Around . . .

Above all, DPS 1100 is an exceptionally powerful tool for improving programmer productivity. And that improvement can be quantified in three ways:

- in terms of the lines of code which *do not* need to be written and debugged
- in terms of the *reduced* time needed to develop and finish a program
- in terms of the *lesser* time needed to make modifications to an existing screen interaction program.

Interactive screen development via DPS 1100 is designed to offload burdensome tasks from the programmer, and to automate the complicated and error-prone programming steps involved in screen formatting.

Beyond that, the DPS 1100 screen handler reduces the total programming job by offloading terminal input and output directives, and by providing a high degree of terminal independence.



Increased End-User Satisfaction . . .

Your end users will be even more pleased with your transaction processing system, through the benefits of DPS 1100.

For the first time, the end user can see, early in development, exactly what the system interface will look like. And the end user can even participate in the definition phase itself, because the development of the screen format is a literal entry at the CRT terminal, which can be instantly displayed for review or change without involving the application program.

With DPS 1100, you can tailor your CRT terminal interface to your exact

requirements, and easily modify the interface based on experience or changing needs. With this power, your end users will benefit from quicker changes and minimum disruption.

Suddenly, your end users will experience improved responsiveness in all phases of data processing.

And your entire organization will benefit from the reduced time and cost involved in bringing online a more meaningful and productive transaction processing system.



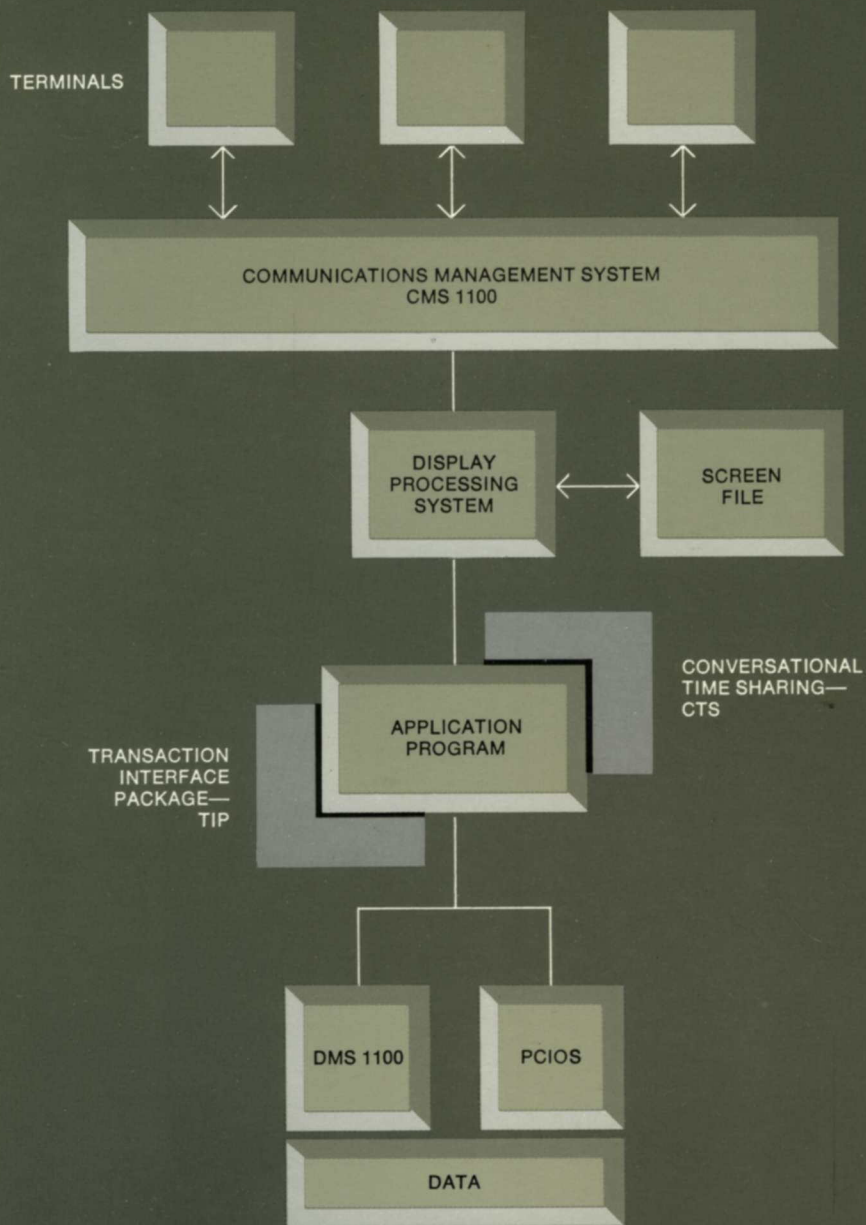


Figure 4: DPS 1100 in a SPERRY UNIVAC Series 1100 Transaction Processing System.

SPEERY  UNIVAC

The computer people who listen