A Guide To Microcomputers For Criminal Justice

Report of work performed under Grant Number 81-BJ-CX-K030, awarded to SEARCH Group, Inc., of Sacramento, California, by the Bureau of Justice Statistics, U.S. Department of Justice.
Points of view or opinions stated in this document do not necessarily reflect the official position or policies of the U.S. Department of Justice.

SEARCH GROUP Inc.
The National Consortium for Justice Information and Statistics
925 SECRET RIVER DRIVE SACRAMENTO, CALIFORNIA 95831 (916) 392-2550
DR. ROBERT J. BRADLEY, Chairman STEVE E. KOLODNEY, Executive Director

Prepared by

Miles J. Ennis

Assistant Director

SEARCH Group, Inc.

TABLE OF CONTENTS

PREFACE

SEARCH began work on *Microcomputers and Criminal Justice: Introducing a New Technology* (Technical Report No. 23) in 1976, a year when most actual users of microcomputers were people who read circuit diagrams for fun. Radio Shack was known as a seller of ham radio gear and TV antennas, and Apple wasn't the anything of anybody's eye. Micros were just crossing the line from experimental technology to marketed product. Thus, we were able to provide basic information on microcomputer technology at the time when agencies were beginning to be interested in small computer systems, and Technical Report 23 came to be the most widely requested SEARCH publication.

Subsequent SEARCH small computer activities have included system developments, such as the Jail Administrator's Management System (JAMS); workshops and seminars; and technical assistance to individual agencies. We have also published a series of articles in our quarterly newsletter, *Interface*, discussing practical issues associated with acquiring and using microcomputers.

Preceding page blank

We believe that the process of acquiring a microcomputer is significantly different from that of acquiring a mainframe computer or even a minicomputer. In fact, the approach traditionally associated with larger computers will lead to unnecessary expense and delay if adopted for micros. This guide suggests an appropriate sequence of steps for buying a microcomputer. We have avoided, as far as possible, detailed discussions of microcomputer technology.

Throughout this guide we make reference to computer hardware and software products by name. We do so by way of illustration, and the references should not in any circumstances be construed as an endorsement of the particular product by SEARCH. Most of the computer and software names mentioned in this book are manufacturer's trademarks and should be treated as such.

1 INTRODUCTION

"We just bought a computer. Tell us what we should be doing with it."

With minor variations, SEARCH receives this particular request for assistance about once a week from criminal justice agencies around the country. We respond as best we can, suggesting applications and (if the agency has been lucky in its choice of computer) identifying existing software packages that the agency might want to consider.

The problem is that the agency, carried away by the computer revolution, has purchased a solution before it bothered to define the problem. That is why luck becomes a factor and why the agency probably faces unnecessary expenses and delays in putting its computer to work. It is also the reason that there are a number of computers gathering dust in agency storerooms.

Acquiring a microcomputer system should be a rational and organized process, the careful matching of agency requirement with computer capability. Our purpose in this guide is to provide criminal justice agencies, particularly those agencies interested in acquiring their first computer, with a sequence of steps and with basic information that will allow them to establish such a process. The guide is designed for a non-technical audience. We provide a glossary, for it is our experience that much of the difficulty in dealing with computers lies in the unfamiliar, and oftentimes bizarre, jargon of computer people.

The steps that we recommend for the selection and implementation of a small computer system differ from those traditionally associated with larger computers. This difference is shown in the exhibit below.

Figure 1

STEPS FOR COMPUTER INSTALLATION

Large Computer	Small Computer
1. Define requirements.	1. Define requirements.
2. Investigate hardware.	2. Investigate software.
3. Acquire hardware.	3. Select software.
4. Design software.	4. Investigate hardware.
5. Program software.	5. Acquire hardware.
6. Test software.	6. Install system
7. Install system.	

Both sequences begin with the same step, that of defining agency requirements, and both end with an installed system, but the large computer sequence puts hardware acquisition first, while the small computer sequence gives preference to the selection of software.

Probably the best way for an agency to cause itself unnecessary problems in acquiring and using a microcomputer is to convince itself that its requirements are unique and thereby lose the opportunity to benefit from the experience, and the investment, of others. Our experience at SEARCH is that once you have adjusted for differences in nomenclature and for relatively minor procedural variations, the requirements of one agency tend to be remarkably similar to those of others. Difficulties that agencies have had in installing previously-developed software packages have almost always related to problems of matching hardware requirements rather than software requirements. The software the agency wanted would not run on the agency's computer without expensive modification. By selecting the software first and letting it guide the choice of a computer, the agency avoids this problem.

We recommend that once the agency has determined, with as much precision as possible, what it is it needs the computer to do, it seek existing software capable of meeting those requirements. Only then should the agency select a computer--one that is compatible with the software. This is the most important recommendation we make in this entire guide.

If suitable software can be located in the public domain or as a proprietary package, then the selection of computer hardware is narrowed to systems on which the software will run. If no suitable software can be located, the agency should reconsider its requirements before it consider initiating its own system design and programming project for a microcomputer.

Spending $50,000 to $100,000 to develop an application for a $5,000 to $15,000 microcomputer rarely makes sense when that application will be used by a single agency; yet that is the typical range of costs for most application development projects.

The variety of software available for micros is enormous and is growing daily. There are packages specifically designed for criminal justice and packages that respond to the needs that criminal justice agencies share with other kinds of organizations. There are excellent database, report-writing, statistical, planning, forecasting, accounting, personnel, inventory, and communications packages. There are even programs that write programs--the so-called application generators. In short, the odds are good that, if an agency is systematic in its search and is willing to be flexible to a modest extent, it will be able to identify existing software consistent with the its needs.

2 CREATING THE SYSTEM SPECIFICATION

No one ever tried to sell a crosscut saw on the basis that it would be the only woodworking tool that the purchaser would ever need. Saws, like most of the tools we use, have a single purpose. They are highly specialized and, for this reason, it is relatively easy to specify how they will be used. After all, you would not use a saw to drive nails.

Computers are different. Within the broad functional category of information processing, they are general purpose tools. The same microcomputer that can keep track of employee records can also perform a multiple regression analysis and help schedule court appearances. It can even function as a sophisticated word processor or as an intelligent terminal linking into a communications network.

Buying a saw is essentially a one-dimensional decision, but buying a computer, even a modest microcomputer, involves multiple dimensions. The decision can be reduced in complexity, but never to the point that it is as simple as buying a hand tool or even an automobile. In view of the inherent complexity of the decision, it is particularly important that it be based upon a clear understanding of the agency's requirements. *The key first step in acquiring a computer is that of preparing a good requirements specification.*

There are a number of approaches to creating a specification. We suggest one approach in the following pages, but the agency should feel free to establish its own sequence. The important thing is to be systematic. Note also that the completed specification will be used as a standard and a guide for evaluating alternatives, not as a rigid mold into which the system must fit with perfection. There are no perfect computer systems. Even those which are custom-designed for the agency will involve compromises and be only an approximation of the ideal.

Who should be involved in developing the system specification? There are two groups: those individuals who have the authority to make decisions about the system and those responsible for operating the system. With a microcomputer system, these two groups may well involve the same people. In fact, it might boil down to a single individual who has both authority and responsibility for the system. The key word in any case is *involvement*. Without early involvement, the specification may be inadequate, because it does not reflect the real problems that the agency must face, and the computer when it arrives may be viewed as a diabolical device rather than a useful tool.

We suggest a five-step process for creating a requirements specification:

(1) Define the problem you want to solve.

(2) Specify the information that the system must provide, if it is to solve the problem you have defined--the output of the system.

(3) Specify the information that must go into the system to allow creation of the outputs you require--the inputs to the system.

(4) Specify the information that must be stored in the system--the files.

(5) Specify security, confidentiality, and audit controls for the system.

Each of these steps is discussed in the following pages.

Define the Problem

As we noted in our introduction to this section, the computer is a tool. In itself, it does not solve problems. That's the job of the people in the agency. All the computer can do is enable them to do the job more effectively, assuming that the computer is the appropriate tool. For example, if a police department has a problem with its officers submitting incomplete or inaccurate reports, a computer won't help. That's a problem of supervision, not of information processing. On the other hand, if the problem is one of making better use of the information that is submitted by officers in reports that are complete and accurate, then a computer may be able to help a lot. Thus, one function of this problem definition step is that of making sure that the computer is an appropriate tool.

This initial step builds a border around what you want the computer to do. It forces you to focus on specific needs, to stop thinking about the computer as a universal panacea and begin thinking in terms of both capabilities and limitations. In this regard, we would caution about expecting a microcomputer to solve a large number of unrelated problems. After all, these are "small" computers. From a cost-performance perspective, they are very cheap and very powerful, but they are most effective when they are dedicated to a single application or, at most, to two or three applications. It is best to justify the computer on the basis of its assisting in the solution of one problem and then add other applications later if there turns out to be sufficient capacity.

We suggest that your problem definition should be very brief--a page at most and preferably two or three sentences. You will be adding plenty of detail as you work through steps 2 through 5. Keep the problem statement uncluttered, so that everyone involved (including potential vendors) will have clear understanding of what you are trying to accomplish.

Define System Outputs

Once you have defined the problem, your next step is to decide just what information you will need your system to provide. A system is useful to the extent that it provides you with information that is relevant, understandable, accurate, and timely. No matter how much information is stored in the computer, it is of no value unless you can get at it when you need it.

System outputs can be categorized as:

- Inquiries are displays (on a video screen) or printouts (on paper) of information that is contained in a single record. Usually an inquiry output is in response to a specific request, such as a driver's license number or crime report.

- Listings are displays or printouts of information from multiple records, based upon some selection criteria, i.e. "warrants outstanding more than 90 days" or "arrests during the month of April."
- Summary Reports combine information from multiple records to show levels of activity, i.e. "crimes reported by UCR category for 1982." Statistical reports, including those which project future activity, can be put in this category. Most summary reports are printed rather than displayed.
- Graphics present summary information in the form of graphs or pictorial exhibits. These can range from simple bar charts and line graphs to complex multi-dimensional colored displays, depending upon the capabilities of system hardware and software.

Any application will normally require several outputs. For instance, a jail inmate tracking system would probably include the ability to *inquire* about the status of an individual inmate, to *list* the inmates currently housed in each cell block, and to *summarize* bookings and releases by category for a selected time period. The system may even include a *graphic* output for projecting trends in inmate population.

When it comes to defining outputs for your system, start with *necessity*: what information do you really need to make good decisions about this problem? It is amazing how much information appears in manual reports and computer outputs that is never used. Creating useless information is more than a waste of resources, it distracts attention from useful information.

Don't assume that every piece of information included in the current manual report should automatically be included in your computer output specification. Automating a manual process gives you a fine opportunity to get rid of informational deadwood. Ask the people who receive each current output to go through them with you item by item and explain just how they use each item. If you can't get a good explanation from anyone, delete the item.

Once you have decided what information is necessary, organize the information into specific outputs. You need not go into great detail in terms of page formats or positioning on the display screen. These are matters that can be left to the software package. Usually, a simple listing of the items of information that are to be included in each output is sufficient.

Finally, determine the frequency of your outputs. You may require weekly, monthly, or annual reports. You may require that certain inquiries be "on demand." The timing of your outputs will have significant impact on both system resources and system operation.

Define System Inputs

Once you know what you must get out of the system, define what data must be entered into the system to create those outputs. Don't make the mistake of collecting every bit of data that you can think of on the grounds that it might come in useful some day. Most of it will never be used, and all that you will have done is clutter up your files and triple the cost of data entry.

Entering data into the system is usually the most critical and costly computer operation. It is *critical* because the value

of system output depends upon the quality of data input: if the data is inaccurate or incomplete, the outputs will be unreliable and misleading. It is *costly* because it is a personnel function rather than a computer function: staff time, even at the clerical level, is expensive in comparison to computer time.

We suggest that you start with each item of information included in an output and trace it back to its source. In some cases a single item of output information will be the result of combining two or more input items. Also, some inputs will be used in more than one output. Note any special rules for transforming input into output, i.e. transforming state offense codes into UCR categories.

For each input data item that you identify, you should specify:

- Source How is this data item created? Is it taken from a form? Who is responsible for its being accurate?
- Timing and Volume How often is this data item collected and what is the total volume on an annual or monthly basis?
- Range of Values Is the item numeric, alphabetic, or a mixture of numbers and letters? Are there only certain values that are acceptable? Does the item have a structure, such as month--day--year?

With this information you can calculate the amount of time necessary to enter data into the system each day by assuming that it will take approximately as long to enter the data as it would to type it. You may discover that the volume and timing of data entry conflicts with the timing of output production. For instance, your application may require a 24-hour inquiry capability that cannot be interrupted for long periods of data entry. In these cases, the agency may be forced to consider acquiring software and a microcomputer system that support multi-terminal operation. Operating two or more terminals simultaneously, however, greatly increases system costs and complexities and should be avoided unless it is absolutely necessary.

You can also begin to establish edit criteria to help keep bad data out of the system. For instance, the range of acceptable values for a data item called "month" would probably be 1 through 12. Anything greater than 12 would be rejected.

Define File Requirements

Information stored in a computer system is usually organized into data elements, records, and files. A *data element* is the equivalent of a single entry on a form, a *record* is the equivalent of the form itself, and a *file* is the equivalent of the complete collection of all the forms of that type.

You can estimate the amount of data that will be stored in the system (and thereby determine the amount of storage capability that will be required) by:

(1) grouping the data elements to be stored into records, based upon the kinds of transactions represented in the system;

(2) estimating the length in characters of each data element and adding those lengths together to determine the approximate length of each type of record;

(3) multiplying the length of each type of record times the number of records of that type that will be stored in the system to determine the approximate size of the file comprised of records of that type.

(4) adding together the calculated lengths of each of the files to determine total system storage requirements.

Most of the above is straightforward, involving common sense and fifth grade arithmetic. However, step (3), determining the number of records of each type that must be stored by the system, can be complex, in that it involves not only an estimate of the number of records coming into the system but also the length of time before they are purged. Allow for file growth and if possible establish reasonable record purge criteria when you prepare the estimate.

Define Policy Controls

Systems operated by criminal justice agencies must have proper security, privacy, and audit controls. Many of these controls will be external to the system itself, consisting of regulations and procedures designed to insure that unauthorized personnel do not have access to the system and that dissemination of information is consistent with privacy regulations and sound management practices.

Control dimensions, however, should be available in the computer software. These include, at minimum, provision for the periodic creation of backup files and an audit log that preserves a record of transactions involving access or changes to sensitive information. Many file management software packages include a security capability for specifying different levels of file access based on passwords.

Security for microcomputers is often less of a problem than it is for larger computers, in that micros are almost never shared with non-criminal justice agencies and, because they are physically small, can be located in relatively secure office areas. (Microcomputers, by the way, have been used as a tool for breaching the security of large telecommunication systems. The ability of the micro to function as a terminal and at the same time have the capability to rapidly process large amounts of data allow it to be employed to try every possible password combination, for instance, or test for chinks in the operating system associated with unusual combinations of instructions.)

It is possible to encrypt sensitive information that is contained in computer files. There are in fact plug-in components available for some small computers that will automatically carry out the encryption process. Note, however, that encryption is a poor substitute for well planned and managed installation security.

At the specification stage it is important to list those control functions that are essential to system operation.

* * *

With your system specification in hand, you are ready to proceed to the next step in the process, that of identifying appropriate software. The specification establishes a reliable baseline against which each alternative can be evaluated. It also provides the agency with an indication of the size of the microcomputer system that will be required. A requirement that the agency maintain files of more than one million

characters on-line would probably force the agency to seek a system that includes a hard disk drive rather than floppy diskettes. The need to be able to access the system from two or more terminals leads the agency to consider systems, both hardware and software, with a multi-terminal capacity. Software is discussed in Sections III and IV of this guide, hardware in Sections V and VI.

3 SOFTWARE PACKAGES

The software packages available for microcomputers can be categorized as shown in Exhibit 2 below. The requirements set forth in your agency's specification probably fall within one or more of these categories. We have also included a discussion of software packages that were created specifically for criminal justice applications.

Figure 2

CATEGORIES OF SOFTWARE

- Statistical
- Word Processing
- Spreadsheet
- Financial
- Database
- Application Generator

We provide estimated price ranges for most of the software categories. As the market has become more competitive,

Preceding page blank

prices have tended to fall and it is possible to find packages that are priced below the suggested ranges. Further, there is no clear relation between product capability and quality and product price.

Statistical Software

Comprehensive statistical packages, capable of performing the full range of standard statistical calculations, are available for most microcomputer systems at prices as low as $30. Most of these packages include capability for creating and updating data files.

Word Processing Software

Word processing is a standard microcomputer application. There are software packages that will perform most of the functions of expensive, dedicated word processors. Since the micro usually does not have the specialized function keys that are built into full-time word processors, using it as a word processor may seem awkward at first and require several day's practice, but most users are reasonably satisfied with the actual performance of the software. There is a substantial variation in features among the available packages, so the agency should be prepared to take the time to identify the package that best meets its specific requirements. Also, if the agency plans to use the micro primarily as a word processing tool, it should probably obtain a formed-character rather than a dot-matrix printer. Word processing software ranges in cost from around $200 to $700.

Supplementing word processing software are programs that will proofread copy for spelling errors. A few packages will even flag phrases like "at this point in time" or "in order to" as containing redundancies. Packages range in price from $150 to $300.

Spreadsheet Software

The automated spreadsheet, which allows the user to set out an array of interrelated data in columns and rows and then manipulate that data, is one of the most popular microcomputer applications. Spreadsheet programs are excellent planning and budgeting tools, in that you can define initial estimates and then explore the impact of alternatives, with the computer automatically taking care of the actual calculations. Most packages provide for printing as well as displaying the spreadsheet, complete with headings and labels. Costs for these packages range from $250 to $500.

Financial Software

General ledger, payroll, accounts receivable, accounts payable, and inventory software packages are widely available. These are geared to the needs of private organizations rather than public agencies. However, many of these packages can meet agency requirements with very minor adaptations. A number of software houses market accounting software as a series of compatible modules, which allows the user to automate one function at a time, beginning perhaps with general ledger and then going into payroll processing or vice versa. Prices range from $200 to well over $1,000 for comprehensive accounting systems.

Database and Application Generator Software

The typical database software package allows the user to define sequences of inputs, specify record storage formats, and specify some small number (usually around 5) of report output formats. Features included in some of the more sophisticated packages include:

- multiple input formats;
- selection of records for pre-defined reports and inquiry searches according to user-specified selection criteria;
- multiple files, including selecting and combining records from several files into a single output report;
- output listings of selected records and summary reports;
- mathematical and statistical processing;
- sort and file merge capabilities; and,
- the ability to interface with a word processing package.

As should be obvious from this list, a full-featured database package comes very close to being a complete, general purpose system. In fact, one variation of the database package, the *application generator*, claims to be just that: a general-purpose package that allows the non-technical user to define his own specialized application in a matter of a few hours.

With database software more than any other category, the adequacy of the user's requirements analysis is essential. These are very powerful software packages, and if congruent with the user's needs, they can provide immediate processing capability.

Criminal Justice Packages

Few software packages have been created specifically for criminal justice applications on smaller microcomputers. With the exception of the PATROL/PLAN software, developed by the Institute for Public Program Analysis to run on Apple II and Radio Shack TRS-80 Model I/III, each of the nationally known criminal justice packages requires either a top-of-the-spectrum microcomputer system, costing close to $20,000, or a minicomputer system costing in excess of $40,000.

PATROL/PLAN, a patrol resource allocation system, is in the public domain and is available from SEARCH for the cost of copying the programs and documentation.

The other established national systems in the public domain are:

- Police Operations Support System--Elementary (POSSE);
- Crime Analysis Support System (CASS);
- Fleet Management Information System (FMIS);
- Investigative Management Information System (IMIS);
- Basic Offender-Based State Corrections Information System (Basic OBSCIS);
- Morgan Crime Analysis System; and,
- the minicomputer version of the Prosecutor's Management Information System (Mini-PROMIS).

With the exception of the PROMIS system, which is managed by the Institute for Law and Social Justice (INSLAW), information about the capabilities and hardware requirements for these systems, as well as copies of the actual software on diskette or magnetic tape and system documentation, are available from SEARCH. In most cases, the only charge to criminal justice agencies is for reproduction costs.

SEARCH also distributes a proprietary application-generator called TAILORBASE, designed to allow criminal justice agencies to define their own application based on their specific requirements. There is a fee for TAILORBASE, and it is usually offered as part of a hardware/software package that includes a microcomputer with a large capacity hard disk and a multi-user capability.

Sometimes, it is possible to transfer an application developed by another agency into your agency. In fact, this is the mode SEARCH recommends for acquiring minicomputer and mainframe software. The SEARCH National Clearinghouse for Criminal Justice Information Systems provides technical assistance for system transfer, including searching an automated index of over 750 operational systems in criminal justice agencies for possible matches. Unfortunately, there are almost no systems available for public domain transfer to smaller microcomputer configurations. Most agencies will be better served to explore the large number of inexpensive proprietary packages.

4 SELECTING SOFTWARE

We recommend a three-stage process for the selection of a software package:

(1) developing a general sense of the full range of alternatives that are available;

(2) performing a detailed analysis of the packages that appear to be most promising; and

(3) scoring those packages that survive the analysis against your requirements specification.

As an inevitable by-product of this process, you will acquire a better understanding of microcomputer hardware. In fact, we recommend that you review Section V of this guide, *Microcomputer Capabilities and Features*, before you start.

Surveying the Market

One of the best things about microcomputer software is that there is so much of it. These riches, unfortunately, can appear overwhelming to the neophyte who is unprepared for the number of products that are showcased in this candy store. One must become a *knowledgeable* neophyte as a necessary first step to becoming an experienced microcomputer user.

This guidebook will give you a start in learning about microcomputer hardware and software, but there are many other sources that you should explore if you are to make intelligent decisions for your agency.

• **Microcomputer Books** Today, many general bookstores carry at least a dozen books about buying and using microcomputers, and technical bookstores usually have several shelves devoted to the subject (Figure 3). Other books are advertised in the microcomputer magazines. In our experience, most public libraries are not a good place to find books about microcomputers: libraries appear to have a philosophy that a

Figure 3

Microcomputer publications are available at computer stores.

few general (and generally outdated) books about data processing should be enough to satisfy their patrons. Good university libraries are a better source, at least for hardbound books and technical periodicals.

We can offer little in the way of advice for selecting books about micros, other than to suggest that you probably should avoid any that have a publication date more than three years old: the evolution of computer technology is occurring so rapidly that three years is about the maximum time a book can be useful without substantial rewriting. Selection guides to small computers, books about programming all of the popular micros in all of the popular computer languages, and even books on widely-used software packages such as Wordstar and Visicalc are available. Appendix A provides an annotated list of useful books.

• **Microcomputer Magazines** In view of the volatility of the microcomputer market, probably the best source of information about both hardware and software is the micro magazine. The growth of these publications has paralleled the growth of the market itself, to the point that *Byte* (the grandfather of the general-interest microcomputer periodicals) is approaching Sears catalogue size. Other magazines that devote substantial space to business (as opposed to home or hobbiest) applications of microcomputers are *Creative Computing, Microcomputing* (formerly Kilobaud), *Desktop Computing,* and *Interface Age.* In addition, each of the extremely popular microcomputers, i.e., the Apple, the Radio Shack TRS-80 series, the Pet, and the IBM Personal Computer, has at least one magazine that is devoted exclusively to it.

All of these magazines publish reviews of software packages and computer hardware. With the exception of their obvious general enthusiasm for anything having to do with micros, the reviewers appear to be reasonably objective. Sometimes the letters to the editor columns are particularly valuable in calling attention to flaws in products or failures in vendor support. Most of the magazines provide reader service cards that are a convenient means of obtaining additional information about products advertised.

- **Computer Shows** Trade shows that feature microcomputers are becoming increasingly common in large and even medium cities. These shows not only provide an outstanding opportunity for you to talk to vendors, pick up their literature, and see demonstrations of their products, but also to attend brief seminars and workshops. Admission fees usually are low. Often, you can obtain free tickets from your local computer dealer.

- **Training Programs** Each year SEARCH sponsors a few workshops on microcomputers in criminal justice. In some states training is available through the Police Officer Standards and Training (POST) programs. However, most of the training available for micros is not oriented to criminal justice. Lack of a criminal justice orientation is not a real problem, however, for the circumstances of law enforcement, courts, and corrections agencies are not that unique. In fact, the main thesis of this guide is that agencies are able to use software packages developed for private industry. The same holds true for training.

In seeking computer training, we strongly recommend that you begin by checking out the programs available through colleges and universities, either regular classes or extension programs. Not only are these courses usually less expensive than privately-sponsored workshops and seminars, they also provide more opportunity for learning: at least at the introductory level, you can cover a lot more ground in a class that meets regularly for a quarter or a semester and includes out-of-class assignments than you can in a two or three-day workshop, no matter how professional and dynamic the workshop presentation.

- **Computer Stores** Contacts with the staffs of your local computer stores during this phase may well signal the beginning of a transition into the next phase of the software selection process, during which you will narrow the choice down to a small number of packages. Chances are that you will eventually acquire your system either from a computer store or a specialized software house, which is for all practical purposes a computer store disguised as a consulting and programming firm. We advise against buying essential software or hardware by mail. *The agency must have someone nearby who can be held responsible in the case of problems.*

There is one basic rule in dealing with a computer store: *the representative of the store should be able to respond to every question you ask in a manner that you can understand.* Taking refuge in jargon at the expense of plain English or in glittering generalities instead of simple facts is a sign that store management is either incompetent or indifferent to your needs. The burden should be on them to make you understand the alternatives they offer. Don't be afraid to ask lots of questions or to keep asking the same question over and over again until you get a satisfactory answer.

Narrowing the Choices

During the process of surveying the microcomputer market you will probably have identified several software packages that look like they might match your requirements. You may even have extended the specification to add features that you did not think of when you conducted the requirements analysis. By this stage you should be able to say something like, "We need a database management package with a solid output capability, including the ability to produce monthly summary reports. The program planning requirement can probably be handled by one of the spread sheet packages."

It is now time to focus on a small number of alternatives, to develop a thorough understanding of their capabilities and special features. You may want to narrow in on one or two vendors, either computer stores or systems houses, if it is possible for you to do so within the constraints of your agency's procurement procedures.

For each software package that you have identified as a possibility for your agency, we recommend the following:

1. Obtain Complete Documentation The quality of the documentation that is available for a software package does not necessarily mirror the quality of the software itself. People who develop software often have trouble *speaking* standard English, much less writing it. A software package can have documentation that reads like a third grader's explanation of

quantum theory and still perform well, if you can figure out how to get it to perform at all.

What the quality of the documentation does reflect is the commitment the vendor has to its customers: well-written and understandable user's manuals show that the vendor is willing to put itself into the user's shoes, to anticipate the questions that the user will have, and then go to the considerable expense of translating the answers to those questions from computer jargon into plain English.

Many vendors, recognizing that intelligent users review documentation as part of the selection process, make their software manuals available for a modest charge. It also may be possible to obtain the documentation on loan either from a computer store or from an organization that is using the software. If the agency issues an invitation for bids, provision of complete documentation should be one of the requirements. *Be sure to review the documentation before you make any commitment to acquire a software package.*

2. Get a Demonstration One of the best things about buying a software package is that you can see it work before you buy it. In the old days, with bigger computers, the user would tell a systems analyst what he wanted. The analyst would translate the user's request into a system specification that would then be translated by a programmer into one or more computer programs. After several weeks or months, the new system would be unveiled to the user, whose standard response (in the words of T.S. Eliot) was, "That is not it all; that is not what I meant at all." The user's expectations were rarely satisfied, and there were delays and additional expenses while the programs were revised.

Insist on a complete demonstration of each software package that you are considering. As far as is practical, involve the agency personnel who will be responsible for system operation and duplicate the actual data structure, input procedures, and output formats that the agency will employ. See how closely the package can be configured to your specifications. List every point that could require a departure from the specification. Go beyond routine operation to test the ability of the package to recover from problems such as power failures or misuse of the system by untrained personnel. The more rigorous and realistic the demonstration, the more certain you will be about the real capabilities of the software.

3. Get Other Opinions The vendor should be able to provide you with the names of organizations that are employing each software package that you are considering. When you contact these organizations, get as much information as possible about the operational characteristics of the software. Ask them about any problems or limitations they have discovered. Try to get a sense of how well the personnel who have immediate responsibility for operating the software like it. Finally, explore the quality of support that the vendor provides: how long does it take them to respond to a problem? Do they have the knowledge and skills to resolve the problem themselves or do they refer it to someone else?

Just because one organization has had trouble with a software package does not necessarily mean that the package is not for your agency. You need to develop a feeling for the general competence of the user organization and for their willingness to give the package a reasonable trial. Some excellent pieces of

software have been used as a handy scapegoat to cover up other basic problems.

Making the Decision

By this stage in the process you should have a clear understanding of the software options that are available to you. You should have narrowed your search to two or three packages that appear to be capable of meeting some significant portion of your requirements. The decision itself may, for all practical purposes, have been made. It is best, however, to confirm your decision or actually make the decision in some systematic way. Governmental procurement regulations being what they are, it is also a good idea to document the process. We outline a sequence for matching agency requirements against software capabilities using a simple scoring technique. If your agency has an established process for selecting from among proposals or bids, that process should be applicable to the task of selecting software.

1. Translate your system specification into a simple requirements list. We suggest that you carry out this translation by recasting the specification as a series of "must" or "should" statements, e.g. "The system must be able to produce monthly summary reports showing caseload activity." or "The system should allow the supervisor to obtain a video display showing month-to-date workload totals." "Must" statements are those that are absolutely essential to the ability of the software to respond to agency requirements. Failure to satisfy any one "must" requirement would in itself disqualify a software package. "Should" statements, on the other hand, represent desirable but not essential features. They are features that are nice to have, that would increase the value of the software to the agency, but which could be gotten along without.

Add to your list whatever statements relating to documentation, vendor support, training, etc. that are necessary to reflect the findings of your general survey of software packages. These statements, too, should be categorized as musts or wants, depending upon their significance to your agency.

Finally, separate the musts and wants into two checklists.

2. **Evaluate the importance of each optional feature (each "should") and assign it a weight.** We suggest that you select the statement from the "should" list that you consider most important and arbitrarily assign it a weight of 10, then go through the rest of the list, assigning weights between 0 and 10 to the other items, based upon their significance in comparison to the item you selected as most important. It is possible that several items will be assigned the same weight.

3. **Match each software package that you are considering against the list of essential features (the "must" list). Disqualify any package that does not satisfy every requirement on the list.** Any package that survives this step should be capable of meeting your agency's essential requirements.

4. **Score each remaining software package against the items on the optional feature list.** Again, we suggest that you use a scale of 0 to 10. If a package is outstanding with regard to a particular feature, assign it a 10 for that feature. If it does not provide the feature, assign it a 0. Multiply this assigned score by the weight previously assigned to the feature

to obtain a weighted score, e.g., a score of 7 and an assigned weight of 6 would be combined to give a weighted score of 42 for that particular package on that feature. Total the weighted scores to determine the overall score for each software package.

5. **Select the software package.** Ideally, the software you pick will be the package that meets all of your "musts" and has the highest total score on your "shoulds." If it turns out that no package qualifies, you are faced with a problem of revising your requirements or reopening the search. There may be no available software capable of meeting your agency's needs, in which case you are faced with a decision on the value of having appropriate software developed. You may wish to consider investigating a larger computer.

We are convinced that the vast majority of agencies that look for a microcomputer software package are going to find one. In some cases it may take a combination of packages to cover all of the agency's requirements. Once the agency has made its decision about software, it can then press onward to the more glamorous part of the process--the selection of a microcomputer system.

5 MICROCOMPUTER CAPABILITIES AND FEATURES

It is possible to buy a microcomputer for less than $100 or to spend close to $20,000. At the lower end of the scale, micros are difficult to distinguish from sophisticated calculators, and at the upper end it is virtually impossible to distinguish them from minicomputers. Some micros are marketed as home systems, for games and education; others, as serious business systems. Some store programs and data on cassette tapes, others on flexible diskettes, and still others on large capacity hard disks. Some provide full-color displays, others black and white or green or even amber. The choices seem endless.

Once an agency has defined its requirements, however, and has identified software capable of meeting the majority of those require nents, the choice tends to be considerably narrowed. Many software packages are available for only a single brand of computer, while other packages will run on several brands. Some agencies will need to store and have continual access to relatively large volumes of data or to be able to operate two or more terminals, which of course would tend to rule out smaller diskette-based systems.

This section discusses features of microcomputer systems, focusing on factors that represent real alternatives for the agency. We avoid, as far as possible, the esoterics of microtechnology. For instance, every micro includes a set of "registers" within its processor. Registers hold instructions for internal processing--program addresses, numeric constants, intermediate arithmetical results, etc. Some micros have more registers than others, and each organizes its registers in its own peculiar way. Microcomputer buffs are capable of arguing endlessly about the relative merits of one scheme over another, but from the standpoint of the average agency it doesn't make any difference. From the agency's point of view, a register could just as well be filled with peanut butter as with the address of the next program instruction.

On the other hand, the distinction between dot-matrix and formed-character printers does make a difference, for the agency's decision between the two will determine print quality and system costs.

We are excluding discussion of that segment at the very low end of the microcomputer spectrum that consists of systems that are exclusively marketed as home systems: the Radio Shack Color Computer, the Commodore VIC, the Atari 400, the Sinclair ZX81, the Texas Instruments 99/4A, etc. These systems have many virtues, but they are not suitable for agency use. They lack the capacity, both in hardware and software, for sustained useful operation in a criminal justice environment. Similarly, we will not discuss the various models

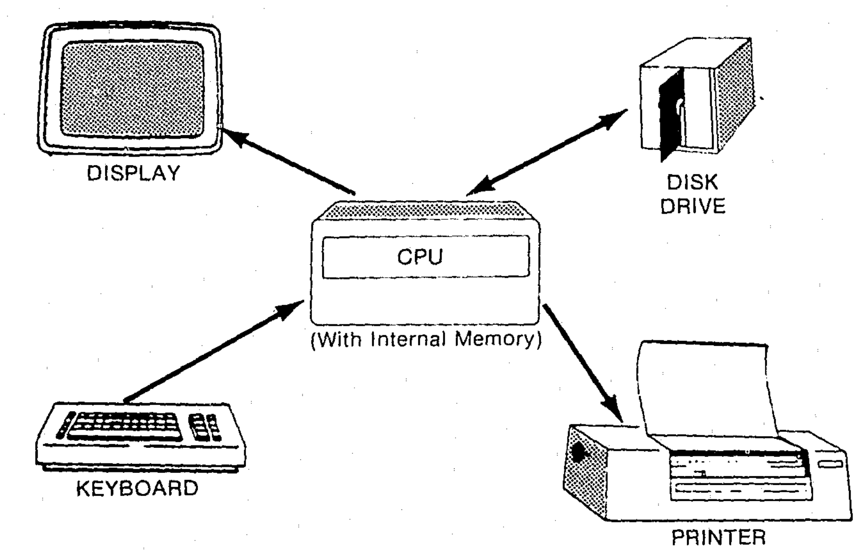
of pocket computers that are now available. Again, these are useful devices, but they lack capacity for our purposes.

It is more difficult to draw the line at the top end of the microcomputer spectrum, that cloudy realm where the difference between minis and micros seems to lie more in the way they are marketed than in their capacities. We shall give a relatively small amount of attention to these more sophisticated systems on the ground that they will less often be acquired by an agency as a first computer.

MICROCOMPUTER SYSTEMS

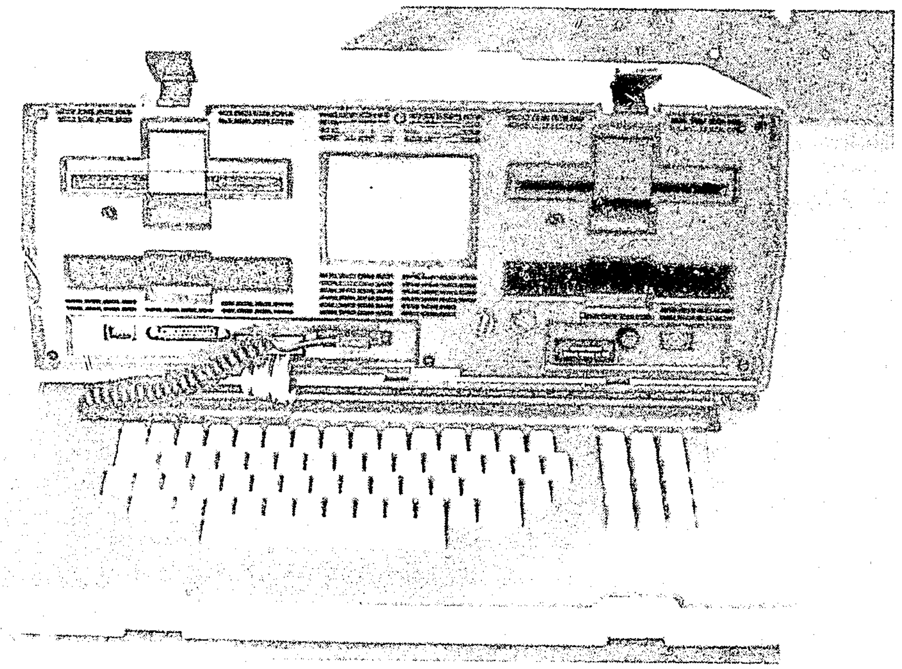
The basic hardware components of a microcomputer system are a microprocessor, internal memory, data storage, a keyboard, a video display, and a printer (Figure 4). Standard software includes an operating system, translators for higher level languages such as BASIC or Pascal, and application

Figure 4



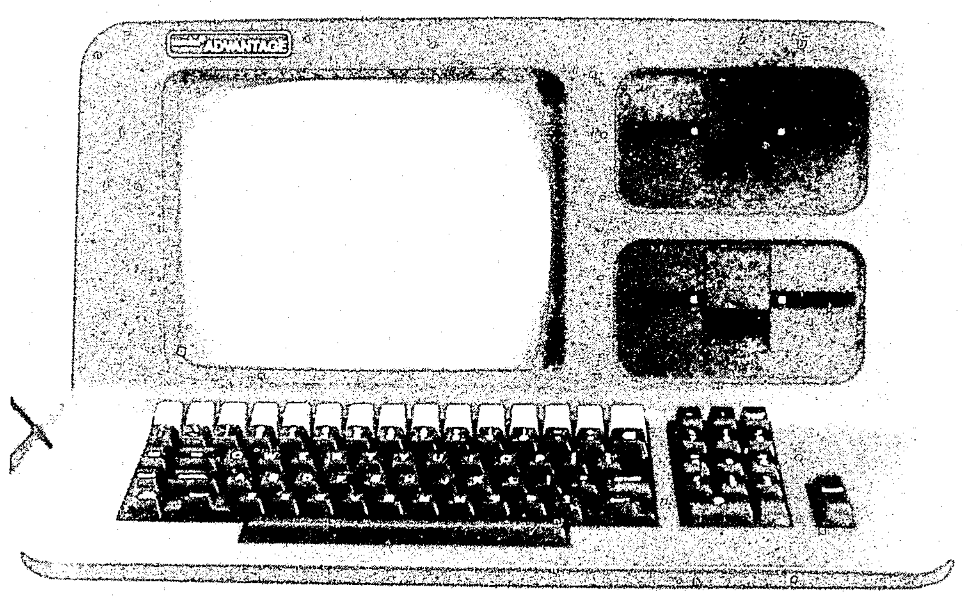
Basic components of a microcomputer system.

Figure 5



A portable microcomputer system.

Figure 6



A self-contained microcomputer system.

programs. The operating system and language software are usually provided with the microcomputer. Application software, as discussed in previous sections of this guide, is selected separately.

Packaging of microcomputer systems parallels that of stereo systems: there are portable systems such as the Osborne (Figure 5); self-contained systems, such as the Radio Shack TRS-80 III or the NorthStar Advantage (Figure 6); and modular systems, such as the Cromemco or Onyx. The particular mode of packaging is not important to the agency. The capabilities of the system are important and are discussed on a component-by-component basis in the following pages.

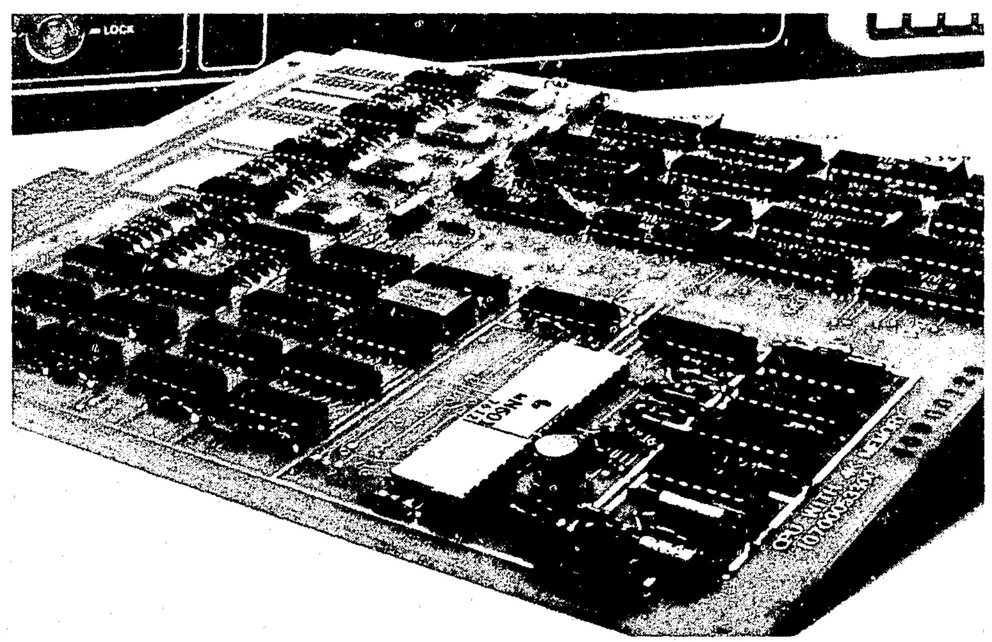
MICROPROCESSORS

The heart of the microcomputer is the microprocessor—the implementation of the functions and capabilities of the central processing unit (CPU) on a single silicon "chip." Each microprocessor incorporates the arithmetic, logical, and control capabilities of larger computers into a single component that is smaller than a dime, but contains the equivalent of several thousand individual transistors and other circuit elements.

Today's personal and small business computers are usually built around an "8-bit" microprocessor, i.e. the standard unit of data that it handles with each operation is made up of 8 bits. There is a trend toward the use of 16-bit microprocessors, which because they handle more data with each operation, tend to be faster and more powerful than their 8-bit cousins. However, the popular 8-bit chips has quite enough power for most small computer applications. In a recent survey of 67 models of upper-end microcomputers, more than 60% were based upon 8-bit chips.

Several manufacturers have marketed systems that employ more than one microprocessor. For instance, the new Radio Shack Model 16 uses a 16-bit microprocessor for internal processing *and* employs an 8-bit processor to handle routine input-output and housekeeping chores.

Figure 7



A packaged microprocessor chip mounted on a circuit board with other components.

The three most common microprocessors are the Z-80, the 6502, and the 8080 families. The most popular 16-bit chips so far are the MC68000, the 8086, and the Z-8000. As with almost anything else having to do with small computers, each has its ardent enthusiasts and equally enthusiastic detractors. Each has strengths and weaknesses, but for all but the most

specialized or demanding applications, the choice of microprocessor is of little importance to the end user, whose concern naturally will be upon the capabilities of the microcomputer system as a complete configuration.

In short, the user should concern himself with what the system can do rather than how it's constructed. Be sure that the microprocessor be from one of the families listed above or at least be widely used and supported.

MEMORY

A computer's internal memory is like the information in a person's head as he or she works to solve a problem or carry out an intellectual task. Such information is of two kinds: (1) information that is the procedure for solving the problem or performing the task and (2) information about the particular problem or task at hand. Example: to calculate approximately how old someone is, it is first necessary to have a procedure (i.e., subtract year of birth from current year) and then to have the information that is unique to the current problem (i.e. year of birth equals 1937 and the current year is 1982). The first category of information, the procedure, corresponds roughly with the concept of a *program* in computereze and the second category with the concept of *data*. Both the program being run and the data currently being processed are present in the computer's internal memory, and the size of the program that the computer can run and the amount of data that can be processed at any given instant are limited by the size of that memory.

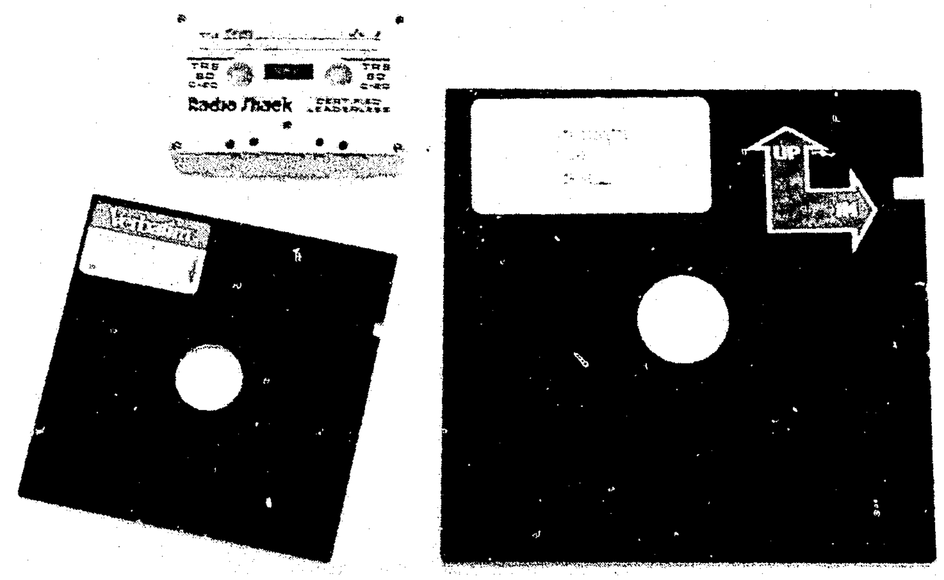
Most 8-bit microprocessors are limited to 65,536 (64K) bytes of internal memory, because that is the maximum that can be directly addressed by the 16-bit memory registers that they commonly employ. In addition, some portion of internal memory is usually occupied by operating system programs, which of course means further limitation on the size of user programs.

We recommend that systems acquired by agencies have at least 32K bytes of internal memory (1K = 1024). In the long run, the agency would probably be better off with 48K or 64K. However, with most systems it is relatively easy to add additional memory, usually in 16k increments, up to the maximum capacity of the micro.

DATA STORAGE

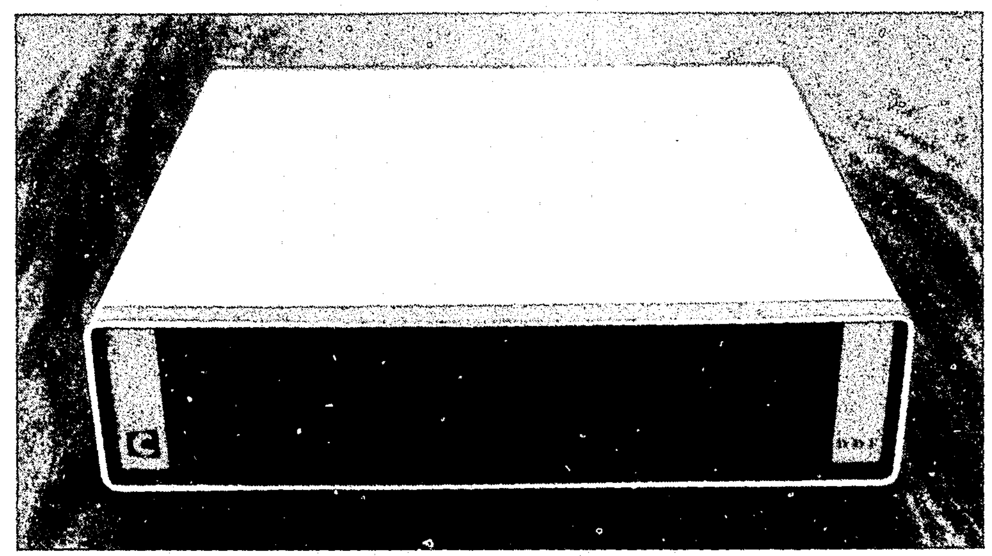
The common data storage media for microcomputer systems include cassette tape, 5 1/4-inch and 8-inch diskettes, and Winchester technology solid disks. Very inexpensive systems employ the tape cassettes, which are no different from the audio cassettes used to record music. These have the virtue of being inexpensive, costing as little as $50 for what is essentially a moderate quality tape player, as opposed to the $300 to $500 cost of a single-density 5 1/4-inch diskette drive. However, they are very slow in terms of reading or writing data and tend also to be quite unreliable. A file that can be loaded in 10 or 15 seconds from diskette may take 3 or 4 minutes to be loaded from this type of tape. (*Cartridge* tape drives, designed for digital comuter systems, are used in a few microcomputers as backup for hard disk; these devices are much more expensive, but also much more reliable and relatively fast.)

Figure 8



Basic microcomputer data storage: tape cassette, 5-1/4" diskette, and 8" diskette.

Figure 9



Two 5-1/2" diskette drives mounted in a separate cabinet.

Tape cassette is not practical for agency applications. Therefore, the decision comes down to floppy diskette vs. hard disk, and that decision should be based primarily upon the size of an agency's files. Data access speed and, of course, cost are also factors. In the survey of 67 upper end microcomputer systems, 31 came with 5 1/4-inch diskettes, 19 with 8-inch diskettes, and 17 with hard disk.

Figure 10

Dual 8" disk drives in a sophisticated word processor.

Diskettes

If your agency's on-line file requirements are under approximately 1,000,000 characters (or bytes), then you can probably use floppy diskettes rather than hard disks. Remember, we are talking about *on-line* requirements, i.e. the amount of data that must be directly accessible to the microcomputer during the time that a program is being run. You may, of course, have any number of additional diskettes containing data

from other files. *We strongly recommend that the agency acquire at least two drives for its system.* Operating with a single drive requires continual swapping of diskettes--a process that leads to annoying delays and, frequently, damaged diskettes. Most microcomputers can handle up to four diskette drives comfortably. The maximum amount of data that can be on-line at any time can be calculated by multiplying the capacity of a single diskette by the number of drives that are available. As with EPA highway mileage estimates, your actual capacity will be lower. Some space is taken up by a directory of the files on the diskette.

Figure 11

Data entry is a key function in any application.

The decision between 5 1/4-inch and 8-inch, as well as single vs. double density, will be partially determined by the file sizes estimated during the agency's requirements analysis and also by the kinds of drives supported by the microcomputers that can run the software the agency has selected. Typical data storage capacities for 5 1/4-inch and 8-inch diskettes are shown in the following table.

Figure 12

Typical Data Storage Capabilities for Diskettes

5 1/4-inch Diskette	Characters
Single Density	80,000-100,000
Double Density	180,000 - 200,000
Double Density/Double Sided	400,000 - 500,000
8-inch Diskette	
Single Density	200,000 - 400,000
Double Density	500,000 - 800,000
Double Density/Double Sided	1,000,000 - 1,200,000

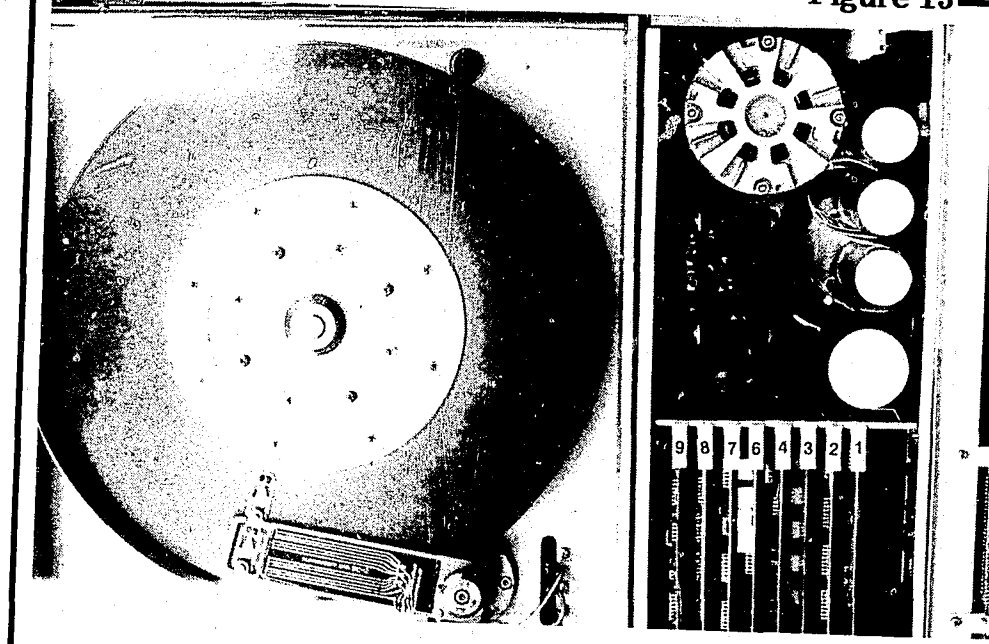
We advise caution in selecting drives that claim an extremely high data capacity--significantly above that indicated in the exhibit. The tendency on the part of manufacturers has been to pack more and more data on a single diskette, but extremes in data storage tend to be associated with problems of reliability, and the agency would probably be better off with drives of proven capability within the indicated ranges.

Finally, diskettes are fragile. They don't take well to rough handling. Agency personnel should follow the instructions provided with the diskettes, and make frequent backup copies.

Hard Disk

Hard disk drives have 10 to 100 times the capacity of diskettes; they tend to be about 10 times faster in terms of reading and writing data; and they are significantly more reliable than diskettes.

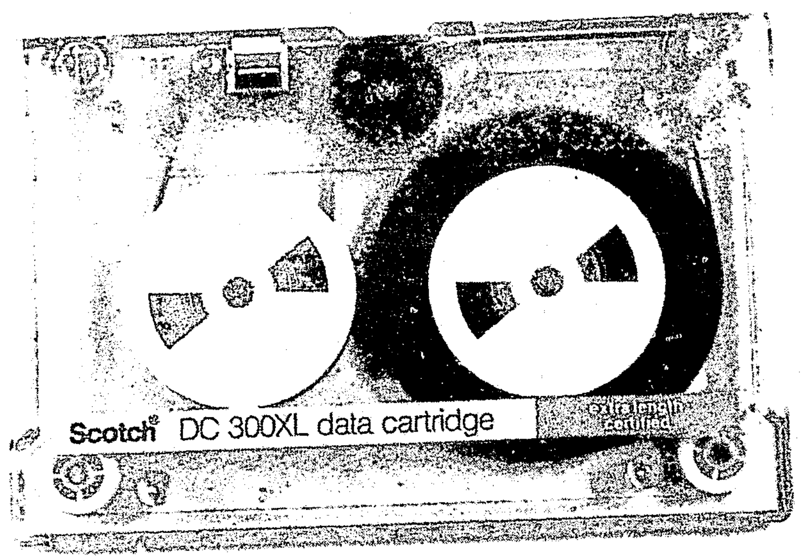
Figure 13



Inside view of a Winchester technology hard disk.

The vast majority of hard disk drives available for microcomputers are based upon what is known as Winchester technology, which means that they are sealed units, i.e. the disk itself is not removable. Thus, it is necessary to provide a second device for backup. Usually this device is either a 5 1/4-inch or 8-inch diskette drive, although some manufacturers use cartridge (not cassette) tape (Figure 14). A few systems with dual disk drives, one fixed and one removable, are coming on the market; however these tend to be priced within the minicomputer rather than the microcomputer range.

Figure 14



Tape cartridge for back-up data storage.

Data capacities for hard disk drives start at around 5 million characters and top out (at least for microcomputer systems) at about 20 million. Drives with greater capacity are marketed, but they represent the extreme edge of the technology and should probably be viewed with caution.

Figure 15

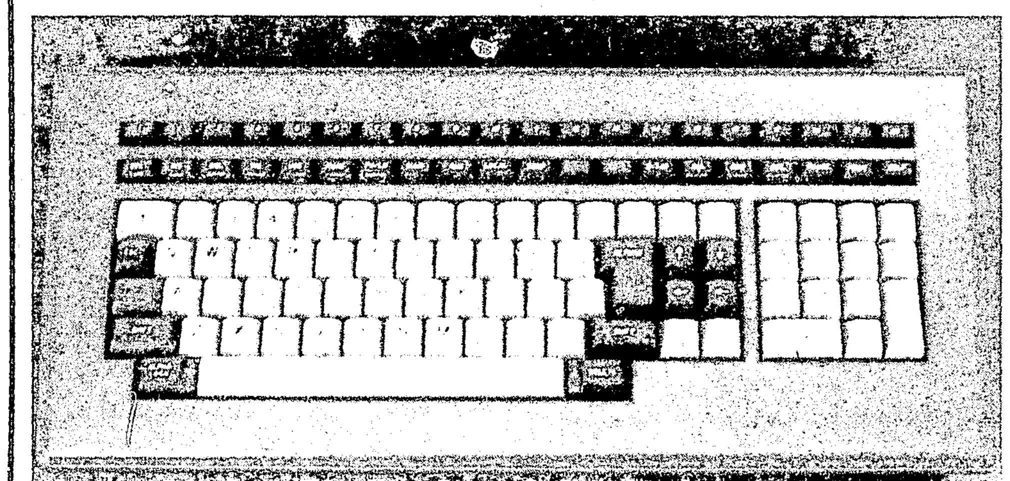
Costs for Diskette and Hard Disk Drives

5 1/4-inch diskette	$300 - 500 per drive
8-inch diskette	$750 - 1,100 per drive
hard disk	$3,000 - 7,500 per drive

KEYBOARD

One of the remarkable aspects of early microcomputer systems was the quality of the keyboards provided: they ranged from barely acceptable to downright useless. Somehow the idea that these systems would be employed for applications that required substantial rapid data entry through the keyboard escaped the manufacturers. It wasn't until the second generation of micros, and in some cases the third generation, that keyboards that were comparable to those of good quality electric typewriters began to be more or less standard.

Figure 16



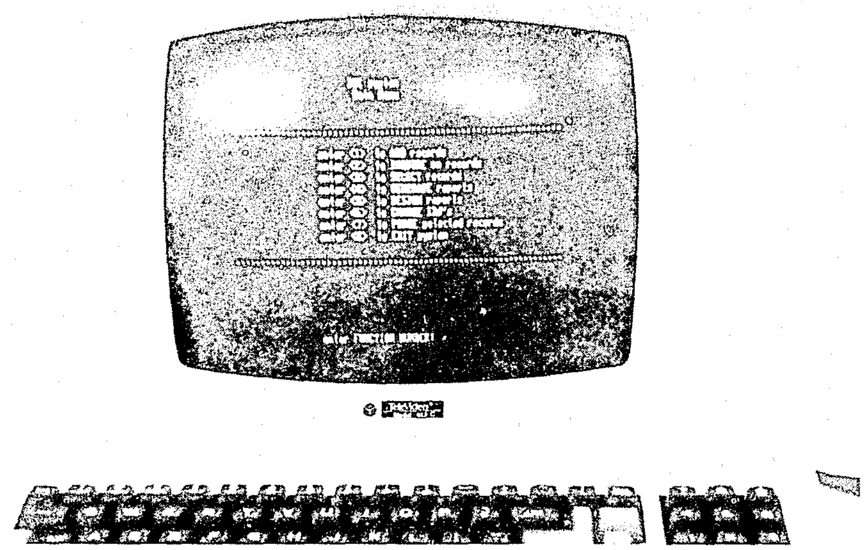
A well-designed keyboard layout that includes programmable function keys and a separate numeric keypad.

The agency should thoroughly test the keyboard of any system that it is considering. *If possible, the test should be performed by the individuals with responsibility for data entry.* Not only should the keys be comfortable to use for long periods of time, but they should be positioned to minimize the possibility of fatal miskeying; e.g. killing the program by accidentally hitting a BREAK key positioned next to the RETURN key.

VIDEO DISPLAY

All micros require a video display. Usually this display is built into the system. However, a number of systems, particularly those built around the S-100 bus, require a separate terminal.

Figure 17



Standard computer terminal displaying main menu selection routine.

As with the keyboard, the critical factor in evaluating a display is that it be comfortable when used for hours at a time. Comfort relates to the sharpness of the displayed characters, their size, and the extent of any glare associated with the screen. There is an unresolved debate about screen color, with a recent tendency toward green screens over black and white. (Some European studies have indicated that green is preferable to black and white in terms of viewing comfort and that yellow is superior to green, but that the differences are so marginal as to be insignificant.)

Other display factors to consider are:

- The number of rows and columns of characters that may be displayed on the screen depends on the size of the screen and of course the size of the characters. For word processing applications, a display of at least 24 lines with 80 characters per line is preferable. Displays that are 64 or 40 characters wide are common.
- The display screen should be at least 9 inches, measured diagonally in the same manner that a TV screen is measured, and 12 inches is preferable. If the characters are well formed, smaller screens of 7 or even 5 inches can be acceptable—at least for applications that do not require long periods of viewing.
- The display should be adjustable for height and angle to accommodate the preference of the user. Microcomputers or terminals that separate the keyboard from the video screen may be preferable.

INTERFACES

An interface links the computer to another device, be it another computer or a peripheral. Virtually every microcomputer provides some interface capability, ranging from little more than that necessary to attach a printer and perhaps a telephone modem, to multiple interfaces for every sort of peripheral imaginable, including esoteric scientific instruments.

Interfaces are described as "serial" or "parallel." A serial interface transmits or receives data one bit at a time. Only a single wire is required to connect the devices, and for this reason it works well over telephone lines. A parallel interface transmits data a character at a time and normally requires at least eight lines between devices and uses more complex electronics. Parallel interfaces are faster than serial.

From an agency's perspective, the important issue is not whether an interface is serial or parallel, but whether it allows the agency to connect its system to necessary devices. The early days of micros were like the early days of railroads in this country, in that there was at first almost no standardization: just as the trains belonging to one railroad could not run on the tracks belonging to another railroad, because there was no established standard for track width, early micros rarely were able to communicate with each other or to be hooked up to previously-developed peripherals, because there were no established interface standards.

Fortunately, the situation is much better today. Among the commonly-used standard interfaces are:

- the Teletype 20mA current loop is the serial interface traditionally used for teletypes or other very slow terminals;
- the EIA RS232C serial interface is widely used in conjunction with modems (supporting telephone communications) and for attaching printers to microcomputer systems;
- the IEEE 488 parallel interface can also be used with printers, but also allows attachment of a wide variety of scientific instruments and other peripherals; and,

- the parallel printer interface, commonly known as the Centronics-interface, is used for attaching printers to the system.

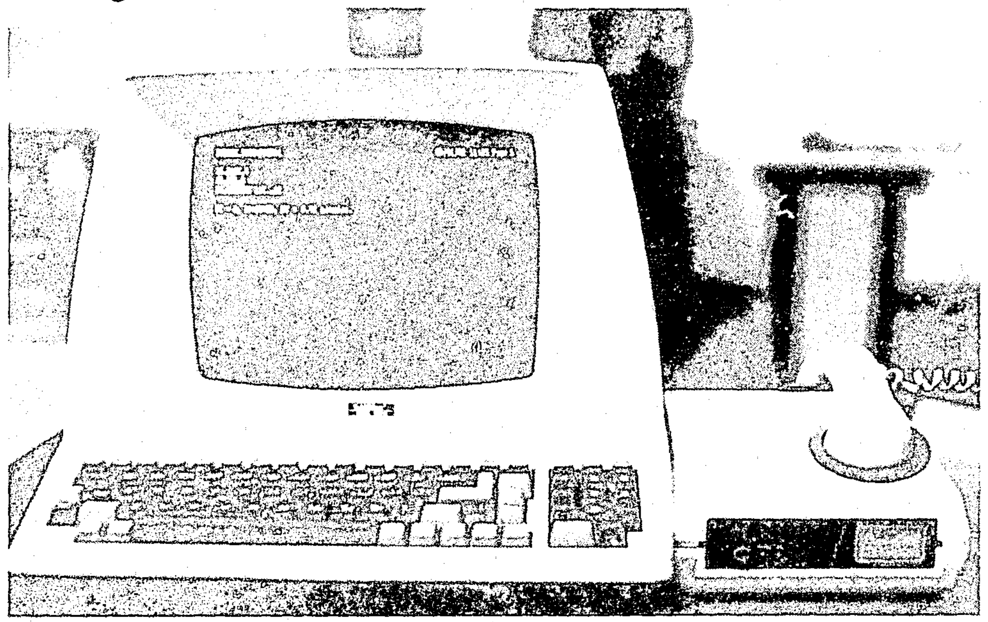
At minimum, a microcomputer system acquired by an agency should provide an RS232C interface and a parallel printer interface. Additional RS232C interfaces are desirable, as is an IEEE 488 interface. The recent trend seems to be away from use of the 20mA current loop.

Modems

In addition to the RS232C interface capability, if an agency wants to link its micro to other computers or to access the reference files of one of the national telecommunications services, such as The Source or Compuserve, it will require a modem. A modem, the word is an acronym for *mo*dulator/*dem*odulator, transforms the digital signal coming from the computer into analog form suitable for transmission over a telephone network and of course transforms analog signals received from the telephone network into digital signals for entry into the computer.

Modems can be characterized as *acoustic* or *direct-connect*. Acoustic modems require that the user dial the number of the computer or network to be interfaced and then insert the handset into the cups on the top of the modem (Figure 18). A direct-connect modem, as one would expect from the name, allows the user to plug the telephone line directly into the modem. (A second line comes from the modem to the telephone, so that the telephone continues to be available for regular conversation when not being used for data

Figure 18



An acoustic modem.

communications.) Direct-connect modems tend to be more reliable and are capable of dialing and answering the phone as well as adjusting their operating characteristics under program control to the requirements of the device at the other end of the line. Acoustic modems are less efficient and can be affected by background noises in the office. Prices for modems range from around $150 for simple acoustic models to around $300 for programmable direct-connect models.

PRINTERS

Printers for microcomputer systems can be divided into two categories, based upon the technique used to create the printed character. One employs pre-formed characters, the other a pattern of dots. Formed-character printers tend to be

slow and expensive, but their print quality is comparable to that of a good office typewriter. Dot-matrix printers tend to be faster and less expensive, but the quality of printing is lower.

There are also very inexpensive printers that employ coated paper or heat-sensitive paper. These produce printing that is of marginal quality and that tends to fade quickly. They are not practical for the majority of agencies. Another option is to convert an office typewriter, such as the IBM Selectric II, to a printer. Converted typewriters produce high quality output, but they were not designed for the sustained high speeds associated with computer printing and tend to be very unreliable.

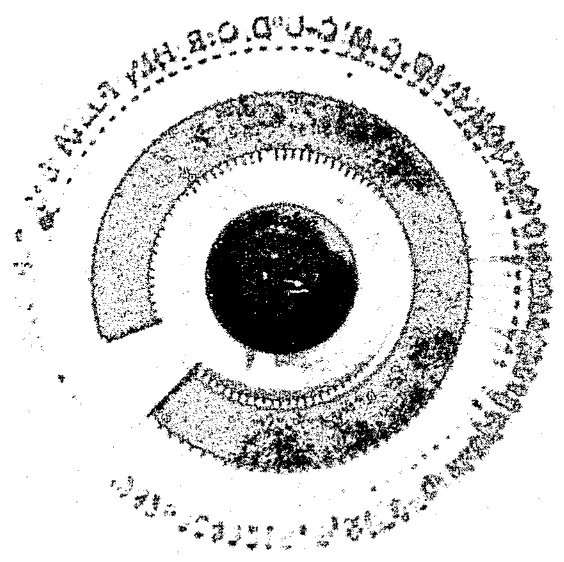
The decision between formed-character and dot-matrix printers comes down to the nature of the agency's requirements. Factors of print quality, speed, reliability, and cost should be considered.

Formed-character Printers

A good quality formed-character printer can produce output that is indistinguishable from that of an office typewriter. Thus, these printers are particularly suitable for word processing applications. Under computer control they can produce boldface print, subscripts and superscripts, underline, and even right and left justify. They can be used to produce camera-ready copy, thereby bypassing the typesetting step in producing documents for publication.

The most common technology for formed-character printers employs a print-wheel, often known as a "daisy wheel" (Figure 19). The characters to be printed are at the ends of 96

Figure 19

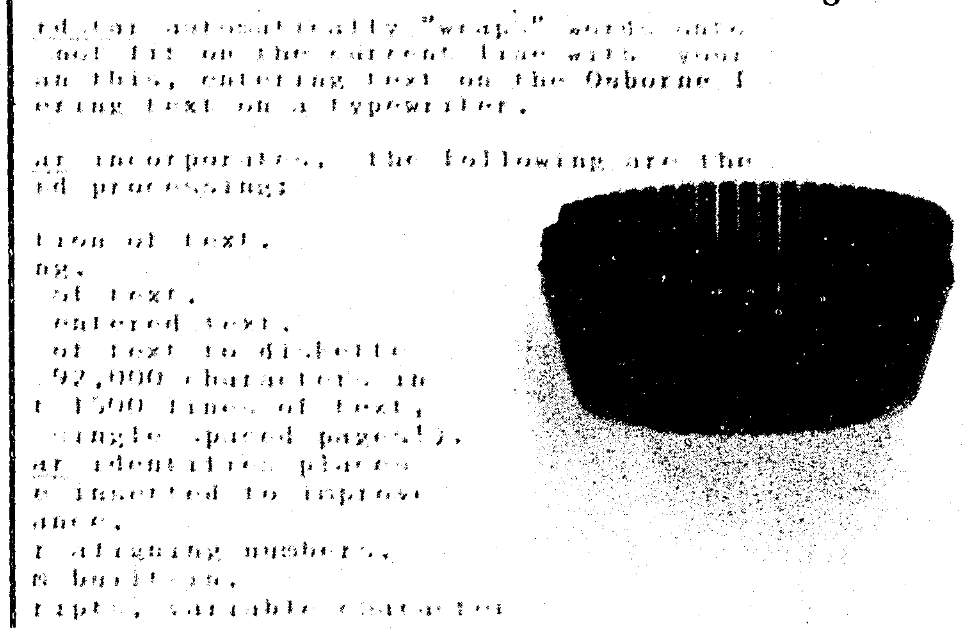


A daisy wheel for a formed character printer.

"petals" radiating from the center. In operation, the wheel is rotated until the proper character is positioned over the paper and then a hammer strikes the end of the petal to print the character. There is sufficient force to produce several carbons. Other techniques employ a type ball, similar to that used on the IBM Selectric, or a rotating thimble-shaped device that is essentially a daisy wheel with the petals folded inward (see Figure 20).

Formed-character printers are slow, with speeds ranging from as low as 12 characters per second (cps) to about 80 cps (see Figure 21). Two techniques used to increase print speed are bidirectional printing and logic-seeking. With bidirectional printing, the print head prints not only as it moves from left to right but also as it moves from right to left, eliminating the

Figure 20



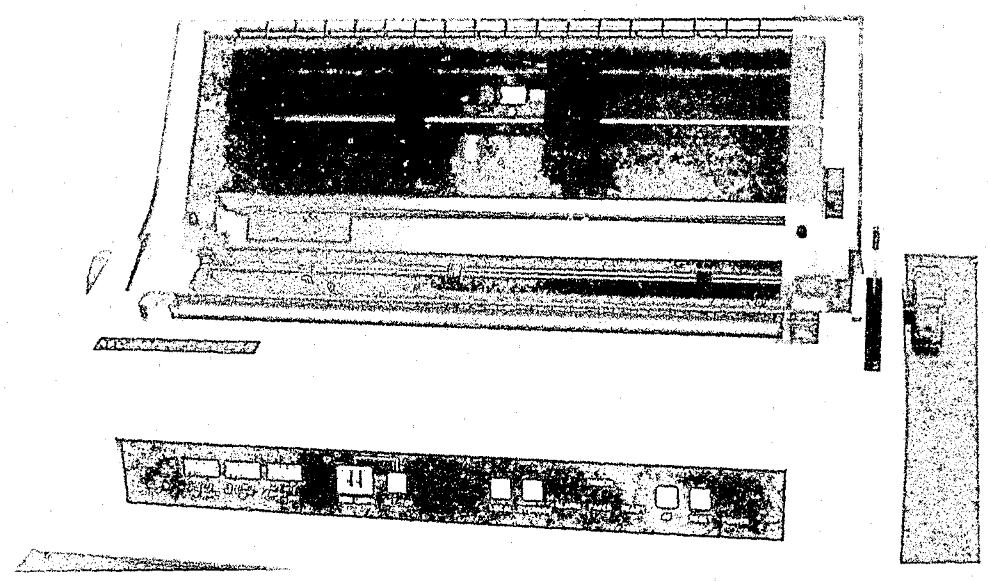
A print thimble for a formed character printer.

time wasted during the carriage return. Logic-seeking means that the printer looks ahead for the last character on the line and, rather than moving the print head over the blanks at the end of the line, immediately skips downward to the next line with characters to be printed.

Many printers also incorporate a memory, known as a buffer. The computer, which is much faster than the printer, can output characters to fill the buffer and then go on to other things while the printer plugs away, printing the characters from the buffer. When the printer is ready for more output, it signals the computer. The size of buffer memory ranges from as low as 1,000 characters up to as high as 16,000.

The three features described above, bidirectional printing, logic-seeking, and a data buffer, are available for both formed-character and dot-matrix printers.

Figure 21



A relatively expensive formed character printer with tractor-feed paper advance.

Figure 22

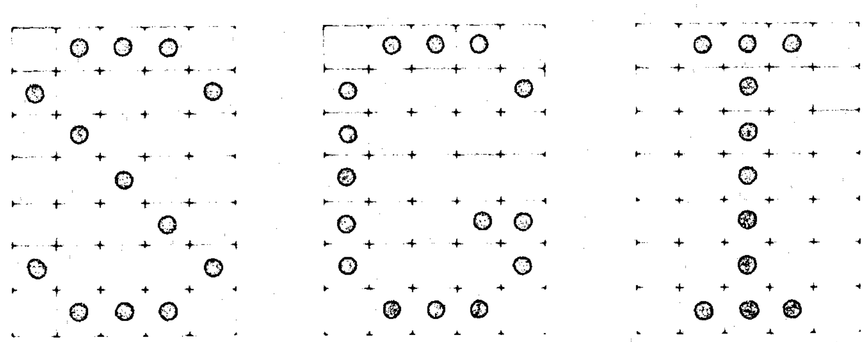


Illustration of character formation using dot matrix.

Faster daisy wheel printers and thimble printers cost in excess of $2,000. A few in the 25 to 40 cps range are available for between $1,500 and $2,000, with printers in the 10 to 16 cps range recently announced for from $800 to $1,100.

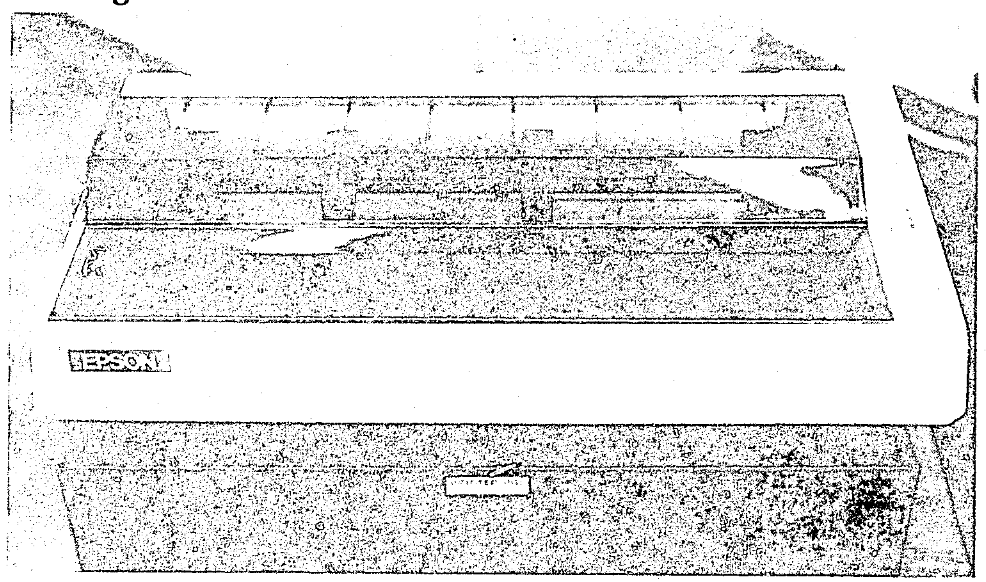
Dot-matrix Printers

Unlike formed-character printers, which are restricted to the particular repertory of characters that appears on the print wheel or thimble, dot-matrix printers have considerable flexibility in forming characters. The basic character format is a rectangular matrix, commonly 5 by 7, 7 by 9, or 9 by 9 dots to a side. Within this format, the pattern of dots that is actually printed (via a hammer striking a series of pins) defines the character. (See Figure 22)

Modern dot-matrix printers can create mathematical symbols, characters from foreign languages, the equivalent of a variety of type styles, and detailed graphics (see Figures 23 & 24). They can produce multiple carbons, as do formed-character printers, and they tend to be both faster (up to 200 characters per second) and cheaper than formed-character. Good dot-matrix printers are available for less than $700.

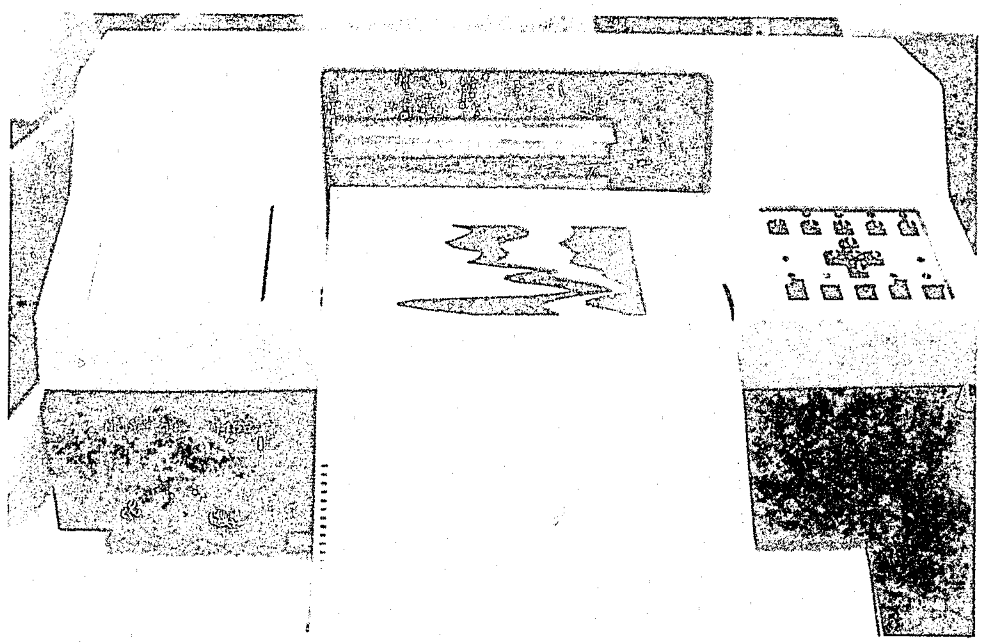
The disadvantage of dot-matrix is that the quality of printing is not up to that of most formed-character printers. It is possible, by striking each dot twice with a slight shift of the print head, to improve the quality of the printing almost to that associated with formed-character, but with a corresponding loss of speed. Many newer dot-matrix printers give the user the option of chosing fast moderate-quality printing or slower higher-quality printing.

Figure 23



An inexpensive dot matrix printer.

Figure 24



A dot matrix printer with advanced graphics capability.

Dot-matrix printers are available with the same bidirectional printing and logic-seeking features found on formed-character printers. Most dot-matrix printers also have a special graphics option that may have value to agencies whose requirements include production of charts or exhibits.

The exhibit below provides a brief comparison of formed-character and dot-matrix printers in terms of print quality, speed, cost, and graphics capability.

Figure 25

MICROCOMPUTER PRINTERS

	Formed-character	Dot-matrix
Print quality	Good to excellent	Fair to good
Print speed	10 to 55 cps	75 to 160 cps
Cost	$800 to 2,000+	$400 to 1,200
Graphics capability	Very limited	Good

Additional factors that should be investigated by the agency are the number of characters that can be printed on a single line, the nature of the paper feed, and the overall reliability of the printer. These factors apply to both categories of printer.

Printers are generally available in maximum line widths of either 80 characters or 132 characters. The 80 character printers are geared to standard 8 1/2 by 11 inch paper, although most are adjustable to handle smaller sizes of paper. The 132-

character printers are geared to the size of traditional computer print-outs. We recommend that agencies obtain the 132-character size, because of the flexibility it provides in producing larger charts and tables.

There are also two general types of paper feed available: friction feed and tractor feed. Friction feed is similar to the paper feed on regular typewriters. It can be used with individual sheets of paper or with rolls. Tractor feed, which is usually more expensive, requires paper with sprocket holes along the right and left sides that are similar to the holes along the edges of photographic film. The paper is moved by pins that fit into the sprocket holes. The major advantages of tractor feed is that it allows you to position the paper with precision—a capability that is of special value to agencies whose requirements include production of printed forms. A number of available printers are capable of both friction and tractor feed.

Finally, printers are the most mechanical components of the microcomputer system. As a result, they tend to have more problems than other parts of the system. The agency should investigate the mechanical quality of the printer it is considering, paying particular attention to the strength and fit of component parts. It is probably a good idea to determine whether a loaner printer can be made available during any period that the agency's printer is away for repairs.

SYSTEM SOFTWARE

In Part 2 of this guide we discussed *application* software—the set of programs that is selected or developed by the agency

to meet its specific requirements. A second category, *system software*, is required by every computer system. System software includes the computer's operating system and one or more language processors. Usually a complete system software package is provided with the computer by its manufacturer or seller. The user of the computer may elect to modify, supplement, or replace the software that is provided as standard equipment.

Operating Systems

The role of the operating system is similar to that of a yardmaster in a freight yard: it controls allocation of system resources, assigning input/output channels and devices, keeping track of available memory, and managing details of file access. Most operating systems also include utility routines, such as editors, sort and merge routines, and special restart procedures.

Today, the microcomputer manufacturer has the option of either developing its own operating system, which would then be unique to its computers, or adopting an operating system that already exists, thereby making its computer more or less compatible with all of the computers from other manufacturers that use the selected operating system. Both options are being used. Apple, Radio Shack, and Xerox, for instance, have elected to develop their own proprietary systems. Osborne, NorthStar, and Onyx have based their systems on existing operating systems: Osborne and NorthStar use the CP/M operating system, while Onyx uses Oasis.

From the agency's perspective, the particular operating system does not make a lot of difference—as long as it is capable of supporting the software the agency has selected and as long as the operating system itself is widely used. This second requirement means, in case of a proprietary system, that the host computer should be one that has already been installed by a large number of organizations or that will be widely installed. The IBM Personal Computer, for instance, is new on the market, but it is reasonable to predict that it will soon become one of the more popular micros.

The most popular non-proprietary operating system is Digital Research's CP/M. Well over half of the 67 upper-end microcomputers described in a recent survey employ CP/M or its multi-user variation, MP/M. Another popular operating system for more powerful micros is OASIS, from Phase One Systems. One benefit of these non-proprietary systems is that micros using them sometimes can run application programs that have already been developed for those operating systems. In the case of CP/M, for instance, literally thousands of programs are available. One note of caution: different models of microcomputer, even within a single manufacturer's line and even employing the same operating system, may not be totally compatible. Make sure the program runs on *your* system before you make a commitment to it.

Language Processors

In this computer age, almost everyone knows that the true language of the computer is a long string of 0's and 1's--something like a list of the scores of every hockey play-off in history. Human beings, at least those who have a normal social life, are not about to learn to speak computer, so its up to the

computer to learn to speak human. The closest that we have come to that goal so far is what is known as the "higher level computer language." There are a large number of such languages: FORTRAN, COBOL, Ada, ALGOL, Forth, Pascal, and BASIC. The language of choice for microcomputers is BASIC, for the same reason that English is the language of choice in San Francisco: it is the language that almost everyone speaks.

There are better languages than BASIC, just as there are languages that are better than COBOL for mainframe computers. Pascal, for instance, almost forces the programmer to be logical and organized in his thinking, and programs written in Forth tend to be very fast and efficient, but every micro comes with BASIC as it standard language and most people who write programs for micros use it.

There are minor differences among the versions of BASIC that are available, but really nothing of serious concern to the agency. Every version that is widely available today (and we argue throughout this guide that an agency should never consider anything that isn't widely available) is competent and none is significantly superior to the others.

SUMMARY
MINIMUM SYSTEM FEATURES

- The system may be based on either an 8-bit or a 16-bit microprocessor. However, it is desirable that the microprocessor be one that is widely used and supported, such as the Z-80, 8080, 6502, or 6800 among the 8-bit machines or the MC68000, 8086, or Z-8000 among the 16-bit machines.

- The system should have at least 32K of internal memory (1K = 1024). For most applications, 48K or 64K is preferable.
- Data storage should consist of at least *two* diskette drives or a single hard disk with appropriate backup. Single drive diskette systems are awkward for many applications.
- The system or associated terminal must have a full-size professional keyboard. The keyboard should be configured to minimize system problems caused by miskeying. It is desirable that the keyboard include a set of programmable special function keys.
- The video display should be at least 9 inches (measured diagonally), be adjustable for viewing angle, and present images that are comfortable for several-hour-a-day use.
- The system should provide at least an RS232C serial interface and a Centronics-type parallel printer interface. An IEEE-488 interface is also desirable.
- The system should include a printer capable of producing a full 132-character line. The printer can be either a dot-matrix or formed-character, depending upon the nature of the agency's applications, but it should be capable of withstanding the relatively-heavy printing requirements of agency use--as opposed to personal or home use.
- The software operating system should be one that is widely used, either because it is provided as a standard feature by the manufacturer of one of the more popular microcomputer lines or because it has been accepted as a non-proprietary system that is available for a large number of microcomputer models. At the moment, selection of a non-proprietary operating system should be limited to CP/M, MP/M, or OASIS.
- The microcomputer should include a full-featured BASIC language interpreter. If the agency has experienced COBOL programmers on its staff, a COBOL compiler could be useful.

6 FACTORS IN SELECTING A MICROCOMPUTER

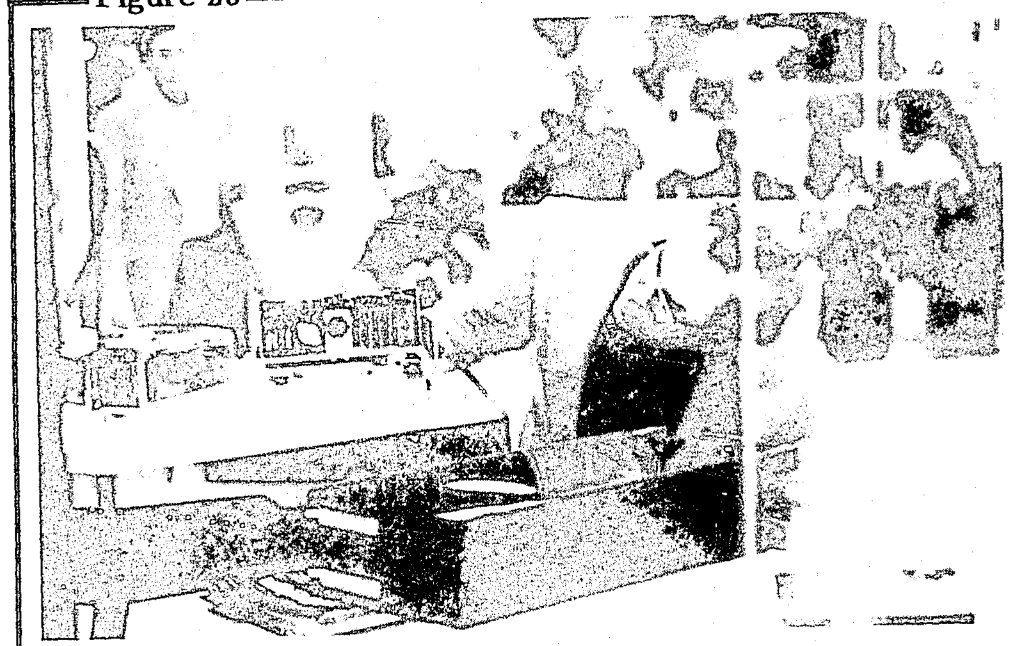
Once the agency has identified one or more software packages that will satisty its needs, it can focus its investigation of microcomputer hardware upon those systems that are compatible with the software packages. In a few rare cases the agency may have no choice at all, in that the selected software will only run on one microcomputer. However, in most cases, the agency will have narrowed its search to a few micros rather than precluded the need for a search entirely.

Section V of this guide provided recommendations about desirable system features. These recommendations define a minimum system configuration. The chances are that any system that met these minimums would be reasonably satisfactory to the agency. Suppose that the agency has identified two or more system configurations employing different microcomputers, all of which satisfy the recommendations. How should it then proceed to make its final selection?

A number of important considerations are not directly related to system features. These include the variety of

Preceding page blank

Figure 26



Selecting a microcomputer requires attention to detail.

software packages that are available for the system, the potential of the system to be expanded in response to evolving agency requirements, the reputation of the system as a proven product, and the availability of maintenance and repair facilities for the system. Training, delivery schedule, and price are also important. Each of these factors is discussed in the following pages.

In selecting hardware, we recommend that you establish a formal evaluation process similar to that recommended in Section IV of this guide for the selection of software packages.

Software Availability

The initial requirements that prompt the agency to acquire a microcomputer should be sufficient to justify its purchase. With time, the agency may discover that the system has

unused capacity or that there are other tasks that should be given priority. It is important, therefore, that a good variety of software packages be available for the selected microcomputer. Other things being equal, the micro with the best selection of software for your agency is to be preferred.

Be cautious, however, in burdening your system with additional tasks. The virtues of the microcomputer lie in its providing accessible computer power at modest cost. As more and more tasks are added to the system, scheduling can become a problem and problems with reliability can increase. The agency may be well advised to consider acquiring an additional microcomputer as an alternative to expecting too much from its original system.

System Expansion

You can't transform a microcomputer into a mainframe, just as you can't transform a Volkswagon into a Mack truck. It is useful, however, to have some room for growth in response to increasing workloads or new applications. Unfortunately, the minimum configuration that we defined in Section V is in fact the maximum configuration for some microcomputers: there is no possibility of significant expansion. If you anticipate that your agency will want to upgrade its system, you will want to select a system that has that potential.

Some of the more common ways to expand the capacity of a microcomputer include:

- **Adding more memory.** Most 8-bit micros are limited to 64K of main memory, and that is sufficient for most users. Some of the newer systems, such as the mid-range IBM Personal

Computer, which has a maximum memory capacity of 256K, and the top-of-the-line Voyager II computer from PCE Systems, which has a capacity of 2,000,000 bytes, allow considerable expansion beyond 64K. These larger configurations allow the agency to run larger programs and to have larger amounts of data actually in memory during program operation. Some of the upper-end micros provide for partitioning of memory, so that more than one terminal can access the system at the same time. (See adding multi-user capability, below.)

- **Adding data storage.** Our minimum configuration recommends at least two diskette drives or a single hard disk with appropriate backup. Hard disks, which have a capacity of 5,000,000 bytes and up, constitute a huge increase over floppy diskettes. As the size of the files that the agency wants to keep on-line grows, additional diskette drives or a hard disk can be added. Most of the popular micros can handle at least four diskettes, and hard disk drives are now available for most models.

- **Adding a printer buffer.** The printer, whether dot-matrix or formed-character, is the slowest component of the microcomputer system. The microcomputer can send data to the printer at a rate of thousands of characters per second, but even the fastest dot-matrix printer can only print at around 200 characters per second. As a result the computer spends a lot of time waiting for the printer, time that could be devoted to productive tasks. Because of this problem, a new microcomputer accessory has appeared on the market--the printer buffer or intelligent interface. These devices, which may be either stand-alone units or circuit boards mounted inside the printer, are essentially limited-capacity dedicated microprocessors with an associated random access memory. Memory size ranges from 16K to 64K. With these accessories, output for the printer is loaded by the micro into the buffer memory until the buffer is full or the micro runs out of characters to be printed. The device then takes control of the printing process, allowing the micro to go about other business. Once the contents of the buffer have been printed out, the device signals the micro that it is ready for more. These devices, which range in price from around $150 for a 16K circuit board to over $500 for a 64K stand-alone unit, would appeal to agencies that have discovered that their micros are having trouble meeting processing and printing schedules.

- **Adding multi-user capability.** As noted above, it is possible to configure a top-of-the-line microcomputer so that it can simultaneously accommodate several users. Depending upon the capacity of the system, as many as seven can be hooked into the micro at one time. Note that we are now in the microcomputer equivalent of the wild blue yonder, that we are talking about a system that has the capacity and demands upon management of a mainframe computer of a very few years ago. About the only thing that a micro of this capacity doesn't require that the mainframe did is its own separate air conditioning system. We are no longer talking about an agency's first computer, and any agency contemplating a need for this kind of capacity should be sure to include minicomputers as well as upper-end micros in its search.

System Credibility

There are obvious risks in acquiring equipment from a vendor that is hardly past its first birthday. Today's Wunderkind may well be tomorrow's casualty, and that multi-talented microcomputer that your agency bought with such high expectations may end up gathering dust in the orphan equipment home. On the other hand, refusing to buy equipment from a company that isn't old enough to vote would exclude some of the best developments in current information technology. In making your choice about a microcomputer, an important peripheral, or even a software package, the best course is to go behind the advertisements, the salesman's claims, the demonstrations and documentation, even the endorsements of other customers, and take a hard look at the company itself.

Does the firm have sufficient capital to weather the ups and downs of the marketplace? Does its staff possess the technical and managerial skills necessary for survival? Is it clearly committed to supporting as well as selling its products? Obtaining answers to these kinds of questions will require some effort on your part, but the effort will be worthwhile.

There are also a couple of rules of thumb that are useful:

- Stay away from products that have not been in the field for at least a year or are not in full scale production. The pride of owning a Chimera I microcomputer with Serial Number 002 can be ingloriously shattered as the engineering change notices begin to arrive or you call the company and discover that the number now gets you Silicon Valley Hot Tubs.

- Never commit your agency to a product that has not been released. This rule is obviously included in the preceding one, but it is worth special attention. A motto of the microcomputer industry could be, "Promise the customer anything, because if we can't deliver today, chances are someone will come up with the necessary product tomorrow." There is a reality here, founded in the recent revolutionary pace of development in information technology, but when you are dealing with a specific product, you can't guarantee the revolution. When the salesman gives you a solemn promise that the new, upgraded model will be out in September, mentally add two years and take a look at the alternatives.

Maintenance and Repair

The ability of a vendor to maintain and repair its products can be an important selection factor. Microcomputers are extremely reliable, the peripherals that support them less so. Printers and disk drives in particular require regular attention, because they incorporate precision moving parts.

The ideal situation is to have a local repair capability. Having your system hung up while you wait for a key item to be repaired and shipped back to you from Osaka can be frustrating. Accordingly, the agency should check the ability of each prospective vendor to handle repairs. Sometimes it is a good idea to arrange a maintenance contract, although with lower-end micros it is often cheaper to pay for maintenance and repairs on an as-needed basis. On occasion a vendor may be willing to provide you with a loaner while some part of your system is being repaired.

Training

Expect the vendor to provide a limited amount of training for agency personnel--two or three days at most and usually in the vendor's shop rather than on the agency's premises. It is usually possible to obtain additional training for a fee, and some vendors actually provide classes on a regular schedule for their customers.

Delivery

Most microcomputers and their associated peripherals are off-the-shelf items if they have been on the market long enough

Figure 27

Involve everyone in the training process.

to warrant agency consideration. The delays of weeks or even months that are typical for delivery of larger systems are rare in the microcomputer industry. Thus, delivery schedule only becomes a factor in unusual cases.

Cost

Cost is always a factor in these times of restricted public budgets, but we list it as the final factor in the selection of a microcomputer because in our judgement cost functions more as a "tie-breaker" than as a primary consideration. Once the agency has identified the software that will meet its requirements; determined the necessary system capacity in terms of memory, data storage, and print quality; and investigated the other factors discussed in this section, the issue of money becomes almost a by-product.

We haven't spoken about the traditional competitive procurement process very much in this guide, because we question its value for the acquisition of microcomputer hardware and software. With the exception of pooled procurements, where the agency may be able to save money by joining with sister agencies in a combined procurement, we recommend a well-researched "sole source" justification as preferable to going out to bid. For many agencies, the documentation of the steps recommended in this and the previous sections of this guide should constitute a foundation for a sole source justification.

Details of the procurement process are addressed in *System Development Guidelines: An ICAP Manual* (see Bibliography).

7 BIBLIOGRAPHY

The most useful book that we know of for agencies seeking to acquire their first computer system is *Business System Buyer's Guide* by Adam Osborne and Steven Cook (Osborne/McGraw-Hill, Berkeley, 1981). Osborne and Cook argue for selecting application software prior to selecting hardware and provide useful guidelines for defining requirements and selecting hardware and software. Their book is resolutely non-technical.

For those who are interested in a more technical approach, we can recommend Osborne's *An Introduction to Microcomputers: Volume 0 - The Beginner's Book* (3rd Edition, Osborne/McGraw-Hill, Berkeley, 1980) and *Your First Computer* by Rodnay Zaks (SYBEX, Berkeley, 1980). Finally, Radio Shack publishes an excellent series of modestly-priced tutorials prepared by the Texas Instruments Learning Center on all aspects of computers. *Understanding Microprocessors* by Don L. Cannon and Gerald Luecke (Radio Shack, Ft. Worth, 1979) is a particularly lucid discussion of the inner workings of microprocessors.

Four SEARCH publications are of particular relevance to agencies planning to use microcomputers. These are:

- *Evaluating Donor Systems: A Software Transfer Technique* (1980);
- *System Development Guidelines: An ICAP Manual* (1979);
- *1980 Directory of Automated Criminal Justice Information Systems* (prepared by SEARCH's National Clearinghouse for Criminal Justice Information Systems and published by the United States Department of Justice); and
- *Local Police Information Policy* (1981).

Each of these publications is available from SEARCH.

Agencies that are particularly concerned about security and privacy should consult *Standards for Security and Privacy of Criminal Justice Information* (SEARCH Technical Report No. 13, 2nd Edition, 1978) and *Modern Methods for Computer Security and Privacy* by Lance J. Hoffman (Prentice Hall, Englewood Cliffs, N.J., 1977).

8 GLOSSARY OF MICROCOMPUTER TERMS

A

Access Time. (1) The time interval between the instant at which data are called for from a storage device and the instant delivery is completed, i.e., the read time. (2) The time interval between the instant at which data are requested to be stored and the instant at which storage is completed, i.e., the write time.

Analog-digital Device. (A-D Device) An electrical circuitry device which converts analog signals to digital information for observation or computer processing.

Addressable Cursor. A feature of some video data terminals which allows the computer to sense the position of the cursor on the screen directly by row and column coordinates.

ALGOL. (Algorithmic Oriented Language). A procedure-oriented high-level programming language used more in Europe than in the United States. ALGOL was the first true procedure-oriented language. (See High-Level Language.)

Alphanumeric. Pertaining to a character set that contains both letters and digits, and usually other characters such as punctuation marks. (Synonymous with "alphameric".)

ALU. Arithmetic Logic Unit. The component of a Central Processing Unit (CPU) which controls arithmetic operations.

Algorithm. The precise procedure for the step-by-step solution of a problem, particularly one that involves repetitive operations or calculations, with all steps and logical actions precisely defined.

Analog Data. Data represented in a continuous form, as contrasted with digital data represented in a discrete (discontinuous) form. Analog data are usually represented by means of physical variables, such as voltage, resistance, rotation, etc.

Application. The specific use to which an organization puts a computer for the accomplishment of some organizational objective.

Application Software. Computer programs which are concerned with the solutions of a user's problem, as with a data processing function, as contrasted with operating a computer itself. (See System Software, Operating System.)

ASCII. (See USASCII.)

Assembly Language. A programming language which provides symbolic mnemonic codes directly corresponding to the machine language codes for a computer.

Asynchronous. Without regular time relationship. Hence, as applied to program execution, unexpected or unpredictable with respect to instruction sequence.

B

Background Processing. The automatic execution of lower-priority programs when high-priority programs are not using the system resources. Also, execution of self-contained (batch) programs, concurrent with on-line, or interactive programs operated from terminals in the "foreground". (Contrast with "foreground processing".)

BASIC. (Basic All-purpose Symbolic Instruction Code) A high-level programming language used primarily on mini- and micro-computers, and commercial time-sharing systems. There is presently no comprehensive formal definition of the BASIC language. Originally developed by Dartmouth University and the General Electric Company around 1963, it now refers to a family of loosely-related languages with considerable divergent features as implemented by various manufacturers and vendors.

Batch Processing. Execution (running) of programs entirely under control of the computer system itself, without human intervention, from a supply of "jobs" placed in the computer from remote or central site devices. In a multiprocessing computer, several "batch" programs (jobs) may be executed simultaneously and independently by the computer.

Baud/Baud Rate. A unit of signaling speed equal to the number of discrete conditions or signal events per second. For example, one baud equals one-half dot cycle per second in Morse Code, one bit per second in a train of binary signals, and one 3-bit value per second in a train of signals each of which can assume one of eight different states.

Binary. (1) Pertaining to a characteristic or property involving a selection, choice, or condition in which there are two possibilities. (2) Pertaining to the numeration system with a radix of two, i.e., in which numbers are written only using the digits 1 and 0 in a positional notation representing powers of the number two as radix.

Bit. Contraction of "binary digit", the smallest unit of information in a binary system. A bit may be either a one or a zero.

Bit Rate. The speed at which bits are transmitted, usually expressed in bits per second (Compare "Baud").

Boards. Assembled units of processors or computer systems comprising a variety of circuitry components. Typical fabrication sizes range from 5" x 7" to 8" x 11".

Boolean Logic. Named for George Boole, 19th Century English mathematician. Operations for combining binary numbers in certain ways corresponding to logical tests and combinations widely used in computer circuit design and in programming for making logical tests.

Bubble Memory. (See Magnetic Bubble Memory.)

Buffer. A storage device used to compensate for a difference in rate of flow data, or time of occurrence of events, when transmitting data from one device to another.

Bus. A conductor, or group of conductors, serving as a common connection for two or more circuits.

Byte. A sequence of adjacent binary digits operated upon as a unit and usually shorter than a word. Typically, there are six or eight bits comprising a byte. Often used somewhat misleadingly as synonomous with "character".

C

Card. (See Board.)

Cassette Tape. An electromagnetic storage medium which utilizes a mylar tape in a fixed container. Typical packaging is identical with cassette tapes utilized in sound reproduction systems.

Centralization. A concept of data processing which provides a single source for processing of several (or many) different information systems.

Central Processing Unit (CPU). A main section of a computer comprising all the circuits controlling the interpretation and logical sequence of instructions (i.e., the Logic Unit), and the arithmetic control and operation (i.e., ALU). In a microcomputer, the CPU is contained on the single microprocessor chip.

Channel. A path along which signals can be sent, for example, data channel, output channel. Also, a semi-autonomous device which controls such data flow (See Data Channel).

Character. Letter, figure, number, punctuation, or other sign contained in text. Besides such characters, there may be characters for special symbols and some control functions. (See Byte.)

Chip. A semiconductor medium comprising thousands of electronic circuitry components in a single small area on a silicon substrate stamped out of a one to two inch circular silicon wafer.

COBOL. (Common Business-Oriented Language) The first business data processing language, and together with FORTRAN, one of the first high-level programming languages. (See High-Level Languages.)

Compile. To prepare a machine language program for a particular computer from a program written in a high-level programming language by making use of the overall logic structure of the program, or generating more than one machine instruction for each symbolic statement, or both, as well as performing the function of an assembler.

Conversational Mode. Communication between a terminal and the computer in which each entry from the terminal elicits a response from the computer and vice versa. (See On-line.)

CPS. Abbreviation for either "characters per second", or "cycles per second", depending on context.

CPU. (See Central Pocessing Unit.)

CRT. Cathode Ray Tube. (See Video Display Unit.)

Cursor. In video data terminals, a moveable indicator which denotes where the next character entered (from keyboard or computer) will be positioned on the screen. The cursor may be a blinking character, a reverse video image, underline, or even be unobservable on some terminals.

Cycle. (1) An interval of space or time in which one set of events or phenomena is completed. (2) Any set of operations that is repeated regularly in the same sequence. The operations may be subject to variations on each repetition.

D

Digital-Analog Device. (D-A Device) An electronic device which converts digital signals into analog values, controlling temperature, current, voltage, frequency, pressures, etc.

CONTINUED

1 OF 2

Daisy Wheel. A print element which is shaped like a small rimless wheel, with character representation on the end of each tine or spoke.

Data. Any representations, such as characters or analog quantities, to which meaning might be assigned.

Data Collection. (1) The act of bringing data from one or more points to a central point. (2) The functions required to assemble data commonly from forms and convert it through an electromechanical device to a computer-readable format.

Digitize. The use of numerical characters to express or represent data. For example, to obtain from an analog representation of a physical quantity a digital representation of the quantity.

Disk. (See Magnetic Disk.)

Disk Controller. A computer system component which commands the disk drive functions including positioning of the read/write head.

Disk Pack. A removable disk unit, typically consisting of several disk surfaces, which can be stored or mounted as needed.

Diskette. (See Floppy Diskette.)

Distributed Processing. A concept of data processing which distributes processing capabilities to several (or many) different locations, typically tied together by communications lines and a central monitoring or limited processor capability.

Dot Matrix. A technique for creating readable characters as patterns of small dots, generally in a 5 x 7 or 6 x 8 matrix.

Double Density. In floppy diskettes or hard disks, a technique of increasing storage by doubling the packing of electromagnetic storage or bits in a given area.

Double-sided. If floppy diskettes or hard disks, a technique of increasing storage capacities by using both sides of the diskette. (Double-sided usage is usually standard with hard disks.)

Dumb Terminal. A video data terminal which has minimal features. Generally such terminals do not have addressable cursors, format protected characters, edit function keys, or special features such as reverse video. (See Glass Teletype.)

Dynamic Memory. A semiconductor memory whose contents slowly fade away, even while electrical power is applied, unless constantly rewritten or refreshed.

E

EIA Interface. A set of signal characteristics (time duration, voltage, and current) specified by the Electronic Industries Association for business machine/data set connections. Officially defined in "Interface Between DP Terminal Equipment and Data Communication Equipment," (RS232, and revisions thereto.)

Emulation. (Emulator) The use of a computer program to mimic (simulate) the actions of a different computer. Thus a PDP-11 emulator would be a program which allows its user to believe he was using a PDP-11 computer. Emulator allows a design team to write programs for a computer which they do not have; the emulated computer may not even have been built yet.

Erasable Programmable Read-Only Memory. (EPROM) A memory type that can be erased with exposure to ultraviolet light and reprogramming by the user an unlimited number of times.

F

Firmware. Refers to the execution of programs from read-only memory, or programmable forms of read-only memory, so that programs seem as much a function of hardware as of software.

Floppy Diskette. A small 8" diameter disk formed of flexible plastic and coated with a magnetic layer for storage of information.

Font. A family or assortment of characters of a given size and style.

Foreground Processing. The automatic execution of the programs that have been designed to preempt the use of the computing facilities. Usually a real-time or on-line, interactive program. (Contrast with "background processing.")

Formatted Data Entry. In video data terminals, the display of a form for data entry with guide words protected so that they cannot be erased or written over. Field lengths are typically specifically defined and cannot be extended.

FORTRAN. (FORmula TRANslation language) A widely used high-level, procedure-oriented language particularly well suited to mathematical applications. FORTRAN was the first "high-level" language developed. (See High-Level Language.)

Free Format. In data entry, the ability to type or submit only those characters and/or fields for which data is present. Padding of extra spaces or zeroes is performed by the computer programs where necessary.

Fuse. An electrical circuit component in PROM's which represents a bit, and can be either blown (a one) or unblown (a zero).

G

Graphics. Refers to the ability of a video data terminal to display lines, both straight and curving, at angles other than horizontal or vertical, and to create arbitrary display symbols.

H

Handshake. Circuitry (also called "handshake logic") which allows control of the transmitting (sending) device by the receiving device during a data transfer. The receiving device tells the sender to pause until it is ready so that action may be taken on the message already received. Handshaking also allows a fast device (like a computer) to transmit data to a slower device (like a printer) without the loss of data by the slower device.

Hard Copy. A printed copy of machine output in a visually readable form (printed reports, messages, etc.)

Hard Disk. (1) Terminology for fixed or rigid disk surfaces as opposed to floppy diskettes. (2) Loosely used to mean a disk device which has much faster "access time" and greater storage capacity than a "floppy diskette".

Hardware. Physical equipment, which makes up a computer system, as opposed to the program or method of use, for example, mechanical, magnetic, electrical, or electronic devices, such as disks, printers, memory storage, tapes, and the central processor of "mainframe".

High-Level Language. A computer programming language which, in contrast to an assembly language, does not have any direct relationship to the hardware instructions (machine language) of the computer. Instead, a high-level language unambiguously expresses procedures, formulas, instructions, computations, logical tests, and operational control by means of a rigorous, well-defined syntax and grammar. A high-level language may be business-oriented, such as COBOL, using English-like sentences and commands, or may be algorithmic, like FORTRAN and PL/1. These programs can be used without major changes on a variety of computers by being compiled (See Compile, Compiler) into machine language for the particular computer.

Host Computer. A computer which provides some level(s) of support to peripheral terminals, intelligent terminals, or smaller computers.

I

Impact Printer. A technique of character printing, whereby the printing is made by physical impact of the typeface against ribbon and paper. An impact printer often refers to fully formed characters, similar to a typewriter typeface in contract to dot matrix printing.

Intelligent Terminal. A device which incorporates a small processor performing some logic functions and interfacing the user inputs and outputs with a host computer.

Interpreter. A program that executes each source language expression before translating and executing into the next one, without first translating the source program into machine language. (See Compile.) Interpreters are slower in execution, but provide greater interaction.

I/O. Input/output. Input or output, or both. The process of moving information into a computer from external devices, such as tapes, disks, printers, terminals, etc.

J

Joystick. An electromechanical device which functions to input x - y vector coordinates to a computer. A joystick usually functions in association with a "Graphics" terminal by manually moving a lever similar to an airplane control stick.

K

Keypad. (1) An input keyboard. (2) A 10 or 12 key input device for entering numerals and the symbols to a computer system.

L

Light Pen. An input device used in conjunction with video data terminals for x - y coordinate inputs, or selection from displayed lists of options.

Line Printer. A printer in which all characters across an entire line of type are printed in one printing cycle.

Load. In programming, to enter data or a program into storage or working registers. (See also Load-and-Go.)

Logging. An audit function which stores on a "log" all transactions or occurrences of specified nature.

M

Machine Language. A language that is used directly by a machine, and corresponds to the actual instruction codes implemented in the central processor logic unit as electrical circuits. A machine language instruction is a binary number of 1's and 0's. (See Assembly Language and High-Level Language.)

Magnetic Bar Reader. A device which reads magnetic bar formations as they pass by a calibrated light source.

Magnetic Bubble Memory. An electronic memory of very high storage density that stores information by changing magnetic polarity of cells or bubbles in a magnetizeable garnet layer.

Magnetic Core Memory. High-speed, random access, read/write memory, made from magnetic core lattices. Core memory is generally faster than other memory technologies, but is also physically much larger and more expensive. It is non-volatile, i.e., data or program contents are not lost when electric power is turned off.

Mass Storage Device. A device having a large storage capacity, for example, magnetic disk, magnetic drum. Also, a high-speed core memory that is external to the central computer memory, and that must be accessed in a manner similar to peripheral devices.

Memory. (See Storage.)

Memory Partitioning. The logical separation of memory areas so that specific programs, or parts of programs, can be stored in them for execution.

Message Switching. The switching technique of receiving a message, storing it until the proper outgoing circuit and station are available, and then retransmitting it toward its destination.

Microsecond. One millionth of one second, i.e., 0.000001 second.

Millisecond. One thousandth of one second, i.e., 0.001 second.

Mnemonic Code. A code that has some easily remembered relationship to the actual word or function, e.g., "LA" for "Los Angeles", and "CLA" for "Clear Accumulator and Add to it from Memory."

Modem. Contraction of modulator-demodulator. A device which modulates and demodulates signals transmitted over communications facilities.

Moore's Law. An empirical relationship proposed in 1964 by Gordon E. Moore of Fairchild Semiconductor who observed that the number of circuit elements in integrated circuit devices (chips) had doubled every year between 1956 and 1964; and who predicted that the trend would continue. Moore's Law has proved accurate (through 1978), but it obviously cannot operate indefinitely. The realization of Moore's Law in integrated circuit manufacture is the main reason for the continual decrease in cost of components.

Multiplexing. A technique enabling the simultaneous transmission of more than one message over a single communication facility.

Multiprocessing, Multiprogramming, Multitasking. These are best defined together, since they have closely related meanings and are frequently used as loosely synonomous, although they are properly distinct. "Multiprocessing" is essentially a hardware-related concept and refers to computer configurations with more than one independent program (CPU) under control of a common software executive or operating system, with each processor running several programs concurrently under the common control. The multiprocessors may share a common memory, or have separate memories running these programs concurrently and independently in order to make optimum use of all the computer facilities (printers, disks, tapes, CPU, memory, communication channels), since no one user is likely to require all such facilities at one time. "Multiprogramming" refers to a computer system which provides for more than one user job (program) into the memory of a single processor. "Multitasking" is closely related to multiprogramming, but properly speaking, refers to procedures or portions of programs (tasks) which operate under a single program identity in an asynchronous manner, i.e., run in time independent or concurrent sequence. A task in this sense is a component, usually of a multiprogramming operating system, concerned with control of a particular device or system resource. Multitasking differs from multiprogramming in that common routines, files, and memory space may be used. Multiprocessing and time-sharing operating systems generally use multitasking to carry out their internal control operations.

N

Nanosecond. One-thousand-millionth of a second, i.e., 0.000000001 second.

Non-volatile Memory. A memory whose contents remain stored when power is removed.

O

OCR. Optical Character Recognition. Machine identification of printed characters through use of light-sensitive devices.

OEM. Original Equipment Manufacturer. (1) A manufacturer of equipment who will sell equipment hardware with specification (and typically some level of software), but without implementation support. (2) A method of procurement whereby the buyer agrees to buy the equipment from a manufacturer as specified. Buyer agrees to be responsible for all interfacing with other equipment as well as all software support. (3) Any vendor who provides "added value" and resells to the end user.

On-line System. (1) In teleprocessing or time-sharing, a system in which the input data enters the computer directly from the point of origin and/or in which output data is transmitted directly to where it is used. (2) Any system in which a user is directly connected to a computer using a typewriter or video display keyboard terminal.

Operating System. Software which controls the execution of computer programs and of all hardware devices that make up a computer system, and which may provide scheduling, debugging, input/output control, accounting, compilation, storage assignment, data management, and related services.

P

Peripheral Equipment. In a data processing system, any unit of equipment, distinct from the central processing unit, which may provide the system with outside communication.

PL/1. (Programming Language/1) A high-level programming language which first came into use in 1966, and which established a range of new patterns and standards for sophisticated languages capable of supporting requirements previously needing assembly language and specialized languages. (See High-Level Language.)

Polling. A technique by which each of the terminals sharing a communications line is periodically interrogated to determine whether it requires servicing. The multiplexor or control station sends a poll which, in effect, asks the terminal selected, "Do you have anything to transmit?"

Process Control. Pertaining to systems whose purpose is to provide automation of continuous operations. This is contrasted with numerical control, which provides automation of discrete operations.

Processor. (1) In hardware, a data processor. (2) In software, a computer program that includes the compiling, assembling, translating, and related functions for a specific programming language--for example, COBOL processor, FORTRAN processor.

Processor Array. A group of computer control processing units (CPU's) or of arithmetic computation units within a single CPU, which operate in parallel, i.e., simultaneously on groups of data of identical characteristics, instead of processing the data one element or group at a time. The ILLAC IV computer is one of the earliest examples.

Program Dump. The listing of the current contents in memory of machine language representation of a program memory area, registers, and other aids so that errors or problems can be inspected.

Programmable Read-Only Memory. (PROM) A read-only memory which is generated in a neutral condition from permanent programming with a customer supplied pattern of ones and

zeroes. PROM's are either programmed by a masking technique as the final step in the semiconductor fabrication, or by the user with a "fuse-blowing" technique, whre each bit is represented by a fuse which is either blown (one) or unblown (zero).

Protocol. (1) An exact description of the code, order, or sequence in which data is transmitted or received between computers or over data lines as in communications protocol. (2) The plan or order of the precise sequence in which computer and humans communicate with each other in an on-line, conversational computer system. The protocol is the order of computer prompting and human response in performing some prescribed activity, such as data entry.

Q

Queue. (1) A waiting line formed by items in a system waiting for service; for example, customers at a bank teller window or messages to be transmitted in a message switching system. (2) To arrange in or form a queue.

R

Random Access. (1) Pertaining to the process of obtaining data from, or placing data into, storage where the time required for such access is independent of the location of the data most recently obtained or placed in storage. (2) Pertaining to a storage device in which the access time is effectively independent of the location of the data. (3) Synonymous with "direct access".

Random Access Memory. (1) Any memory device in which the time to access a particular memory location is independent of the location. (2) A term used to designate semiconductor memory which can be written as well as read, as contrasted with Read-Only Memory (ROM).

Read-Only Memory. A storage device that stores data not alterable by computer instructions; for example, magnetic core storage with a lockout feature and certain kinds of semiconductor (integrated circuit) memory on which the contents are created at the time of manufacture. (Synonymous with "non-erasable storage", "permanent storage", and "fixed storage".)

Real-time System. A system performing computation during the actual time that the related physical process transpires, thereby allowing results of the computation to be used in guiding the physical process.

Record. A group of related facts or fields of information treated as a unit.

Register. A device capable of receiving information, holding it, manipulating or computing with it, and transferring it as directed by control circuits.

RPG. (Report Program Generator) A special purpose programming language for producing complete machine reports from information which describes the data files and the format and content of the output report.

S

Sectors. In disk technology, concentric tracks encircling the disk surface that are logically or physically separated into a number of pie-shaped wedges or sectors by radial lines from the center to the rim. Typically on floppy disks, 32 sectors of 128 data bytes comprise one concentric data track.

Semiconductor. (1) A property of material which provides for a limited conductance of electrical energy. (2) A microelectronic device based on a simple silicon substrate which provides the electrical properties of registers, capacitors, inductors, and transistors in greatly reduced size.

Shift Register. A register in which the stored data can be moved to the right or left.

Signal Processing. Processing of data information received from external (and usually remote) devices. The information may be transmitted by hard wire, by telephone, by broadcast, etc. An example is the processing of data radioed from a satellite.

Simulation. The use of a computer to generate data corresponding to a real event which would be difficult (if not impossible) to obtain upon demand. A computer can be used to construct data to test another computer, another computer program, a human being (example: chemistry experiments "performed" entirely on a computer terminal), or a human-and-computer system. An example of the last case would be the Defense Department's use of computers to "simulate" a war, also known as "simulation gaming".

Software. A set of programs, procedures, rules, and possibly associated documentation concerned with the operation of a data processing system--for example, compilers, library routines, and other programs. Usually distinguished as either system software (concerned with operating the computer itself) or application software (concerned with solving a problem or performing a user function, such as payroll and accounting). (See Application Software System Software.)

Static Memory. A semiconductor memory whose contents remain stable (unchanged) as long as external power is applied, without special refreshing of the contents.

Store and Forward. The interruption of data flow from the originating terminal to the designated receiver by storing the information enroute and forwarding it at a later time.

Synchronous. Occurring concurrently and with a regular or predictable time relationship. (See Asynchronous.)

System Software. Computer programs which are concerned with operating a computer, and not with solving user problems. (See Software, Application Software, Operating System.)

T

Telecommunication. (1) Transmission of signals over long distances, such as the telegraph, radio, television. (2) Data transmission between a computing system and remotely located de vices via a unit that performs the necessary format conversion and controls the rate of transmission.

Teleprocessing. A form of information handling in which a data processing system utilizes communication facilities. (Originally an IBM Trademark.)

Teletype. Registered trademark of Teletype Corporation. (See Teletypewriter.)

Teletypewriter. A generic term referring to the basic equipment made by Teletype Corporation and to similar teleprinter equipment.

Terminal. (1) Any device capable of sending and/or receiving information over a communication channel. (2) In on-line systems, a device with a keyboard and either a printer or a video display screen, used to do programming or data entry and, in general, to display information in the computer and send information to the computer.

Thimble Wheel. A print mechanism which involves a molded print wheel with cylindrical fingers containing the character representations.

Throughput. A measure of system efficiency; the rate at which work can be handled by a computing (communications) system.

Time-sharing. The interleaved use of time on a device--hence, a method of operation in which a computer facility is shared by several users concurrently, each of whom appears to be the only user of the terminal.

Time-slicing. A technique used for supporting many "simultaneous" on-line users of a single "time-shared" computer by giving each one a small interval (slice) of time, usually on the order of 100-200 milliseconds in which he has the computer all to himself, and going in "round robin" fashion from user to user.

Trace. A technique which permits a computer program to generate a historical record of specified events in the execution of a program.

Track. On a disk storage device, one of a number of circular lines of magnetic coating on which data is recorded.

Transistor. A small solid-state, semiconducting device, ordinarily using germanium, that performs nearly all the functions of an electronic tube, especially amplification.

U

UART. Universal Asynchronous Receiver/Transmitter. An integrated circuit enabling a parallel device to independently transmit and receive data along serial (one bit) wires. UART is typically used in the interface between a computer (parallel device) and a cable to a terminal (serial device). The "Receiver" accepts bits one by one from the terminal and assembles them into parallel data for the computer. In contrast the "Transmitter" accepts a parallel code and sends the bits out one at a time to the terminal. Devices connected via a UART operate in an "asynchronous" manner in that the UART "tells" the sending device to pause if the previous data byte has not been taken by the receiving device (see Handshake). "Universal" means that the UART may be set up to handle parallel data of various lengths at various rates, with optional data-checking bits sent along with the data. Such setup may be done with wire jumpers or by control signals from the central processor.

USASCII. USA Standard Code for Information Interchange. The standard code, using a coded character set consisting of 7-bit coded characters (8 bits including parity check), provides information interchange among data processing systems, communication systems and associated equipment. The USASCII set consists of control characters and graphic characters. (Also known as ASCII.)

Utility Program. A program which provides general support for the operation of the computer; for example, a disk copy, program, a memory diagnostic program, or generalized file sorting program.

User. The individual or organization that employs a computer to support the accomplishment of personal or organizational objectives. Users are distinguished from data processing personnel, such as systems analysts, programmers, and computer operators.

V

Video Display Unit, Video Display Terminal. A device which provides a non-permanent display of data and information on a television-like screen by using a high-speed Cathode Ray Tube (CRT).

Volatile Memory. A storage device in which stored data are lost when the applied power is removed, such as semiconductor Random Access Memory (RAM).

W

Word. In computing, a sequence of bits or characters treated as a unit, i.e., retrieved from or stored in memory as a single unit and processed as a unit in the CPU registers. In large computers, word size is usually 32, 36, or 64 bits; in minicomputers, word size is typically 12 or 16 bits; in microcomputers, word size has been 4 or 8 bits, but (1978) 16-bit word processors are becoming more common. A word may be stored at a single memory location (address) or may be comprised of two or more memory addresses treated as a logical unit by the CPU. Thus, in a 16-bit word size microprocessor, the word may be two 8-bit memory locations, i.e., two bytes.

END