# Microcomputer Software: Selection Considerations

ESO 1105

Marvin T. Batte

The author is assistant professor, Department of Agricultural Economics, The Ohio State University.

# Microcomputer Software: Selection Considerations

Computer technology represents a new, exciting, and relatively untapped tool for use in farm business management. Like most emerging technologies, however, it is not always understood by those who can gain by its adoption. My observation is that few managers are in the middle ground of true understanding of the potential this tool holds as a management aid. Those people who fear the computer or who have had little experience with it tend to greatly discount the value of the computer to their business. Those of us who have used these machines only briefly or who have done casual readings in the area of computerization often have over-realistic attitudes of the computer's value. The true and immediate usefulness of this technology probably lies somewhere between these extremes.

The computer has several attributes which give it value as a management aid. Chief among these is the speed with which the computer can process information. It has been suggested that there will be an "information revolution" which will be as dramatic in the future of farming as the industrial revolution was in the substitution of machines and capital for human and animal labor on farms. Agricultural businesses have access to more data than ever in history. The technology exists to create information management systems which will allow automatic collection, analysis and reporting of data created in farm or business activities. Continued innovation and adoption of these technologies will likely occur. This well may speed the current structural change toward fewer and larger farms. With narrowing profit margins, greater reliance on manufactured inputs, and increased use of debt financing, the vulnerability of the present day farmer to price and yield variability has heightened. It will be those farmers who can precisely control costs and productivity and who can most accurately appraise marketing opportunities who will be profitable in the future. In other words, information processing is likely to be of increasing importance as an activity of the farm manager. Computers can be used very effectively in tasks such as budgeting, record keeping, and database management. Furthermore, there will probably be increased use of computers in such process control functions as the monitoring of feed consumption and milk production.

With these introductory statements, let me address the task at hand; to discuss computer software, attributes of useful software, and pointers for those who are at the point of software purchase or development.

## What is Software?

Our first task should be to develop a workable definition of software. As contrasted with hardware which is the physical part of the computer system (printer, monitor, CPU, disk drives, etc.), software is comprised of the instructions and computer programs which make the computer function in a prescribed manner. Software is necessary not only to perform the task defined by the user, but also to coordinate and control the printer, disk storage devices, and other components of computer hardware. It is software that transforms the computer from an expensive piece of desk top clutter into a highly useful management aid. (My remarks here refer to the use of the computer by the business manager, but it should be recognized that in order for the computer to be used for family entertainment or as an educational tool, software is also required.) Because it is computer software that directs the hardware to accomplish useful work, the manager must be well informed about software, and in fact, software availability should be the major determinant of the type of hardware purchased.

There are three major classes of software: (1) the operating system (OS), (2) language interpreter or compiler, and (3) applications.

### Operating Systems Software:

The operating system controls the internal operations of the computer. It is this set of computer instructions which coordinates the peripheral equipment (i.e., secondary storage units, printer, modem, etc.), controls information storage and recall within the central processing unit, provides error messages to the user and many other "housekeeping" chores. The OS typically will be a part of the read only memory (ROM) and cannot be altered by the user. Generally, an operating system will be included whenever a microcomputer is purchased.

One of the primary difficulties with microcomputer technology is the lack of standardization among the various manufacturers. This is especially true of software. The operating system used by a particular manufacturer may be standardized only for computers made by that company, and in some instances, only for particular models of computers. Because applications and language software are dependent on the OS to control the hardware of the computer, applications developed for one operating system cannot be used on another without modification. A useful analogy may be the farm tractor. Here, the tractor is hardware and the driver is software. Most tractors are highly standardized in the way that they operate: That is, they all make use of a steering wheel, brakes, and similar hydralic system controls. Hence, the same driver (software) can likely operate all of the tractors. If we did not have this standardization of controls of the tractor (the operating system software if you will), then the driver would not be able to transfer from one machine to another without additional training.

This lack of standardization is a substantial problem in the computer industry. There are many different computer chips used by the different computer manufacturers. Each of these chips requires a different operating system to control it. The result is that applications and language software cannot be easily transferred among machines. Because this lack of standardization is so troublesome to software writers and users, there is considerable interest in the developmnt of industry standard operating systems. Prior to 1981, an operating system known as CP/M (Control Program for Microcomputers) was quickly becoming the industry standard operating system. This allowed software writers to write applications software using standard CP/M operating system conventions, making it much easier to transfer applications software to other computers. However, in 1981 the IBM personal computer was introduced. It utilized Microsoft Disk Operating System (MS-DOS) rather than CP/M. As a result of the widespread acceptance of the IBM computer, there has been a tendancy to accept MS-DOS as an industry standard. Many computer manufactures are now marketing computers which are IBM compatible or which support the MS-DOS operating system. Because so many computers utilize one of these operating systems, software developers will typically develop software for these systems before developing for other OS. Hence, when an individual is faced with a hardware purchase decision, it is wise to select a microcompurer which can utilize either the CP/M or MS-DOS operating systems. The result is a wider availability of software.

Language Software:

The second type of software with which you should be familiar is that of language compilers and interpreters. This software makes it possible for the user to develop programs written in such high level languages as BASIC, FORTRAN, PASCAL and COBOL. The compiler translates the "high level" language into machine language. Machine language is used for all internal arithmetic and logical functions. This machine language utilizes a binary number code to represent all possible numbers and characters as a series of zeros and ones. To develop a computer program in machine language would be extremely difficult. A high level language compiler allows the user to write the program with a series of easy to understand, English-like statements. The presence of high level language compilers provides the manager with programming experience, or the willingness to learn a high level language a means for development of applications software. Additionally, some applications software available on the market is written in a high level language. In order to operate this software, a high level language interpreter must be available on the computer.

Applications Software:

Applications software refers to a set of computer instructions (i.e., a computer program) designed to perform a particular task or application. These applications could range from such physical tasks as the identification and feeding of cows in the milking parlor to the more commonplace farm record keeping. There are indeed a large number of jobs with which the computer can assist. However, the computer can be used in none of these applications until software is first developed for the particular application.

There are two broad classes of applications software: specific and general purpose. Specific purpose software includes programs designed for the solution of a particular task. Examples would include a farm records program, a dairy health record or the identification and automatic feeding of cows. These programs are useful only for the purpose for which they are designed.

General purpose applications software, as the name implies, can be used for a wide variety of applications. This type software is often refered to as a very high level (or macro) programming language or a programming aid. This software is designed for a particular type of application, but can be modified for the specific application at hand. There are five types of general purpose applications software which come to mind. These are (1.) electronic spreadsheets, (2.) data base management software, (3.) graphics packages, (4.) statistical packages, and (5.) word processing packages. These packages can be used for a large of number of specific purpose applications. For example, the electronic spreadsheet could be used for cash flow analysis, enterprise budgeting, or repetitive mathematical calculations where prices or quantities are to be varied frequently. Data base management software could be used to maintain dairy herd breeding records, herd health records, labor and wage records for business employees or any of many similar applications.

Because general purpose software can be used by farmers and nonfarmers alike, the market for this software is much broader. The cost of the software package will likely be much lower than a specific application package developed solely for agriculture. Due to the ease of use, versatility, and relatively low cost, the manager may find that general purpose applications software will play an important role in the computer system. This packages could well substitute for many other special purpose software packages designed strictly for the farm.

## Sources of Applications Software

There are several sources of applications software for the farmer. These options include (1) design and coding of programs by the manager/programmer; (2) purchase of custom designed software; (3) purchase of existing or prewritten software; and (4) utilization of existing software on a host computer. Associated with each of these options is a series of advantages and disadvantages, some of which are discussed below.

## Manager/Programmer Designed Software

There are a number of farmers who have had programming experience or who are willing to learn a high level language. This opens the door for these people to develop some of their own software. There are indeed several advantages and disadvantages associated with such a choice. Chief among the advantages is that the manager/programmer can design the program specifically for his farm. The manager can specify the exact input available to be used by the program, and the exact form of the output generated. Because the manager must design the program, he will be keenly aware of all assumptions used. The result will be a better understanding of the program and its limitations.

In order to design good applications software it is important to understand the problem itself. If the problem is to develop a least cost ration generator for ruminant animals, the program designer must understand ruminant nutrition. Secondly, in order to efficiently program these instruction, one must be a relatively fluent in a high level language. Errors can result if the designer/programmer is deficient in either of these areas. Also, we must realize that although there is little cash outlay involved with writing a program, it can be a very expensive process. For many programs, large amounts of the manager's time must be committed. If the opportunity cost of this time is high, i.e.,programming takes the manager away from other highly important jobs, then the real cost of program development must be recognized as very high. Once the program is developed, additional time will be required to "debug" the program to insure that it works properly.

## Custom Designed Software

Another alternative is for the manager to define the problem, along with input and output needs and required assumptions, and hire a programmer to design and/or code the program. The advantages of this approach are much the same as for the previous option. The program is designed for the particular farm application, and the manager can participate in program development to insure the program is reasonable and technically correct. Further, if the opportunity cost of the manager's time is high, this option may be much less costly than the previous case even though the cash outlay will be higher.

On the other hand, to include a programmer in the process adds a potentially difficult communication problem. The manager must be able to convey to the programer the exact nature of the problem to be solved and the assumptions to be incorporated. This is substantially more difficult if the programmer does not have a good agricultural subject matter knowledge. For this reason it is suggested that custom program development be done by someone who has at least a working knowledge of the subject matter under consideration. One must also realize that the testing process must be done and, if anything, is more difficult because the manager is not familiar with the program structure and/or code.

Finally, even though this may be a better alternative than the manager developing his own program, it is still a very expensive method.

Purchase of Existing Software

Purchasing existing software is likely the least costly means of owning management software. Because software writers are selling to many producers, they can spread the development costs over more sales, thereby lowering the cost to individual purchasers. In addition, these programs are attractive because they have been debugged and tested before sale, thus largely removing this burden from the manager.

Perhaps the major disadvantage of purchasing existing software programs is that they are not designed exactly for your farm or need. It may not be possible for the manager to determine the exact assumptions used in the program. Even if it were possible to determine how the program should be changed in order to make it more useful for the particular farm, it is often not possible to make changes in the program. This becomes especially important when we realize that technology, prices, yields, and government policies are always changing. Consider for instance the number of management programs which were altered simply because of the changed tax laws of 1981. This is much less critical when the application is not unique to a particular business. An accounting package, ration balancer or income tax calculation program should not require tailoring to the individual farm.

Finally, we must realize that there is not a large amount of agricultural software on the market and that which is available ranges widely in quality, making the selection process very critical. This final point may make a statement for utilizing an agricultural computing consultant when the computer system is designed. Other sources of critical review of agricultural applications software are newsletters such as Doane-Western Agricultural Computing Source Book and the Successful Farming Computer News.

Use of Public Access Software

In a great number of cases it may be wiser not to own software, but rather to use it on a pay per usage basis. There are currently several sources of public access programs. These include such agricultural systems as TELEPLAN (Michigan State University), AGNET (University of Nebraska) and FACTS (Purdue University). These agricultural sources can provide access to excellent programs for purposes such as least cost ration balancing, land bid price calculation, investment evaluation, and the like. In addition, there are several private vendors one can access on a subscription basis, including SOURCE and COMPUSERVE.

Even though the primary advantage of using public program libraries is cost savings, there are other advantages. These programs have typically been developed and tested by subject matter specialists. As a result they should be conceptually correct and error free. Because the software resides on a single computer, it can easily be updated as changes in farm programs or tax policy require. Further, the host university or company can provide consulting services concerning the use of the program.

Again, a key disadvantage of this method is that the program is not specifically designed for the individual farm. This will be of little difficulty for many programs which are not farm specific (i.e., the ration calculation problem). Another problem is that the program cannot be altered by the user. Further, it is difficult to know the assumptions on which the program is developed. Finally, this option requires a small amount of additional equipment to connect your microcomputer to the host computer by telephone.

## Hints for the Purchase of Software

The availability of prewritten agriculture applications software will likely increase dramatically in the next few years. The wide variability in quality of these packages will continue however. For this reason it is very important that the farmer/purchaser exercise caution when selecting from this software. The following hints may be useful.

1. Compatibiliy: Will this software work with the hardware system you currently own?

2. Capacity: Software often is limited in the amount of computer memory that can be addressed. This will limit the size of application. If, for example, a farmer desires a farm accounting system which allows each crop enterprise to be identified separately for each of 10 landlords, then it is important to select an accounting program which allows this many accounts.

3. Coordination: Is there a "coordinated" system available for the needs of this farmer? Coordination refers to the ability of several software packages to tie together, utilizing the same database or allowing exchange of information between programs.

4. User Friendliness: Is the program user friendly and protective? This is to say is the program written in such a way as to help the user operate the program? More importantly, does the program allow for easy recovery from errors? As an example, if an incorrect entry is made by the user, it should be possible to correct this without beginning again.

5. Documentation: Thorough documentation is of extreme value. Is this program well documented as to use and included assumptions? As a rule, if you can't determine the basic assumptions used in the program then you should not base decisions on its results.

6. Program Updates: If there is a tax law change or other major change in the assumptions of the program, will updated versions be available at reduced cost? Some software companies which address highly dynamic topics such as tax law will provide key parameter changes each year in order to maintain the accuracy of the program.

7. Vendor Support: Is the software vendor willing to provide assistance during the period of learning to use the software?

8. Program Modification: Is it possible to modify the program? This will typically require assistance and/or approval from the software writer.

9. Warranty Period: Does the software vendor offer a period of warranty during which program errors will be corrected?

10. Trial Period: Will the software vendor allow a period of use to determine if this software is useful to the purchaser?

## Summary

The computer does indeed have tremendous capabilities for use in the business. To be useful, however, well designed application software software must be available. There are several potential sources of software, each with associated advantages and disadvantages. This software, regardless of the source tends to be expensive relative to the cost of the hardware. Further, there is software commercially available which is of little use, or worse still, is based on conceptually incorrect foundations. The bottom line is that software selection is a very critical process; one which deserves a great amount of exploration and thought. When designing the computer system, consideration should first be given to identification of those tasks for which computer application has the greatest payoff. Next, a search should be made to identify the software which can best meet the needs of the business. Finally, a hardware system should be selected which will support the chosen software.